A MAJOR PROJECT-II REPORT
ON

DETECTION OF ONLINE HUMAN BEHAVIOUR IN HINDI LANGUAGE POSTS BY LEVERAGING MULTILINGUAL BERT.

A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of

MASTER OF TECHNOLOGY

in
Computer science and engineering
by

MISTU MAHAJABIN
(ROLL NO. 2K22/CSE/27)
Under the Supervision of
Prof RAJNI JINDAL
(Prof, Dept of Computer Science & Engineering)

To the
Department of computer science and engineering
Delhi technological university
(formerly delhi college of engineering)
Shahbad daulatpur, main bawana road, delhi-110042. India

May, 2024

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

ACKNOWLEDGMENT

I would like to express my sincere gratitude to all those who contributed to the successful completion of this project. First and foremost, I extend my heartfelt thanks to my supervisor **Prof. Rajni Jindal** and for her invaluable guidance, continuous support, and constructive feedback throughout the duration of this project.

I am deeply appreciative of the assistance and cooperation received from my colleagues and peers who played a crucial role in data collection, analysis, and discussions. Their collaborative efforts greatly enriched the project and contributed to its overall success.

Special thanks to **Prof R K Yadav** for his co-operation and co-ordination which significantly contributed to the successful completion of the project.

I am also grateful to my friends and family for their unwavering encouragement and understanding during the various phases of this project.

Lastly, I acknowledge the support and resources provided by **Prof Vinod Kumar, HOD**, Computer Science Department which facilitated the smooth execution of the project.

This project would not have been possible without the collective efforts and support of these individuals and entities, and I am sincerely thankful for their contributions.

MISTU MAHAJBIN
M. Tech CSE
Roll No. 2K22/CSE/27

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

CANDIDATE'S DECLARATION

I, Mistu Mahajabin, Roll No. 2K22/CSE/27 student of M.Tech (Computer science engg), hereby certify that the work which is being presented in the thesis entitled "**Detection of online human behaviour in Hindi language posts by leveraging multilingual BERT** " in partial fulfillment of the requirements for the award of the Degree of Master of Technology in Computer science Engineering in the Department of Computer Science and Engineering, Delhi Technological University is an authentic record of my own work carried out during the period from August 2022 to Jun 2024 under the supervision of **Prof Rajni Jindal**, Professor, Dept of Computer Science and Engineering. The matter presented in the thesis has not been submitted by me for the award of any other degree of this or any other Institute.

Place: Delhi **Candidate's Signature**

This is to certify that the student has incorporated all the corrections suggested by the examiners in the thesis and the statement made by the candidate is correct to the best of our knowledge.

Signature of Supervisor **Signature of External Examiner**

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

CERTIFICATE

This is to certify that **Mistu Mahajabin** (Roll No. 2K22/CSE/27), has successfully completed the project titled "**Detection of online human behaviour in Hindi language posts by leveraging Multilingual BERT**" under the guidance **Prof. Rajni Jindal.** The project was undertaken as part of the MTech course. Throughout the duration of the project, Mistu Mahajabin has demonstrated exceptional dedication, diligence, and proficiency in conceptualizing, planning, and executing the project objectives.

The efforts put forth by Mistu Mahajabin have resulted in the creation of a comprehensive and insightful project report, showcasing a high level of understanding and competence in the chosen field of study.

This certificate is awarded as a testament to their commitment and achievement in contributing to the advancement of knowledge and skills.

[Prof Rajni Jindal]
[Professor]
[Delhi Technological University]

Detection of online human behaviour in Hindi language posts by leveraging Multilingual BERT

Mistu Mahajabin

ABSTRACT

Hindi is the third highest spoken language in the word with almost 662 million people speaking worldwide. Recent trend shows the increase in usage of Hindi as internet language. When individuals use online platforms to express their perspectives, share knowledge, recount personal experiences, and convey emotions, a significant issue arises when these interactions transform into a space for offensive remarks, comments, and conversations. This project is being built with the aim of detecting antisocial and prosocial online behaviour with a reward or feedback system. The whole project is planned in two phases. In the first phase detection of antisocial online post has been done. Prosocial behaviour and feedback system is developed in the second phase. Antisocial behaviourr is mainly divided into four category abusive/offensive, cyberbullying, Targeted group and hate speech. Except these few more categories are like fake, defamation also there. More than 8000 label data have been used in the both the phases of the project. m**BERT** model with cross entropy loss function and AdamW activation function, was applied. The model achieve maximum F1 score of 0.79. Integration of the two classifiers has been done using weighted average to get a unified score for detection of human behaviuor.

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

LIST OF PUBLICATION

Conference Publications

1.

- **Title**: " Hindi language processing: A survey"
- **Authors**: Mistu Mahajabin, Prof Rajni Jindal
- **Conference**: National Conference on Bigdata Analysis (NCBA - 24)
- **Location**: Kolkata, India
- **Date**: 2nd June 2024**.**

2.

- **Title**: " Performance evaluation of LSA on pilots' qualitative data"
- **Authors**: Mistu Mahajabin, Prof Rajni Jindal
- **Conference**: International Conference on Smart Technology, Artificial Intelligence and Computer Engineering (ICSTAICE-2024)
- **Location**: Delhi, India
- **Date**:09 May 2024

CONTENTS

List of figures

List of tables

List of abbreviation

NLP…….Natural Language processing
BoW……Bag of Words
RF……Random forest
LR……Logistic Regression
BERT……. Bidirectional Encoder Representation from Transformer
mBERT……...Multilingual Bidirectional Encoder Representation from Transformer
EDA…… Easy Data Augmentation
TF-IDF……Term Frequency Inverse Document frequency
CSV…….Comma Separated Values
GPU…… Graphics Processing Unit
CSE…… Computer Science Engineering
DTU……Delhi Technological University
SRS…… Software Requirements Specification
API…… Application Programming Interface

1. INTRODUCTION

1.1 Overview

The internet has become an indispensable part of our daily lives, profoundly impacting how we work, communicate, learn, and entertain ourselves. It serves as a vast repository of information, offering instant access to knowledge on virtually any topic imaginable. The internet has transformed how we collect information, from performing research and utilizing educational resources to keeping abreast of global news and developments. Moreover, the internet facilitates seamless communication through email, social media platforms, and messaging apps, allowing us to connect with friends, family, colleagues, and even strangers across the globe in real-time. Additionally, online shopping has changed the way we shop for goods and services, offering ease and a vast array of options right at our fingertips. Entertainment has also been revolutionized by the internet, with streaming services providing on-demand access to movies, music, and other forms of media. Overall, the internet has become a vital component of our ordinary lives, enhancing efficiency, connectivity, and access to resources in unprecedented ways.

Online systems have become essential in our daily lives due to their convenience, efficiency, and accessibility. From communication to shopping, education, banking, and entertainment, the integration of online systems has transformed how we interact and conduct various activities. One key reason for their importance is the ability to connect people globally, allowing instant communication and collaboration regardless of geographical distances. Online platforms also provide convenient access to information and services at any time and from anywhere, reducing the need for physical presence and enabling multitasking. Furthermore, online platforms provide customized experiences and individualized recommendations according to user preferences, which boosts user satisfaction and engagement. The COVID-19 pandemic further emphasized the significance of online systems by enabling remote work, virtual learning, telemedicine, and online social interactions to maintain continuity in daily activities despite physical restrictions. Overall, the integration of online systems has revolutionized modern living, offering unprecedented convenience and connectivity in our daily lives.

1.2 Motivation

A human behaviour detection system is crucial for numerous reasons in today's society. Firstly, such systems can enhance security measures by identifying unusual or potentially threatening behaviour s in public spaces, airports, or other critical locations. This early detection can aid in preventing crimes or terrorist activities. Secondly, these systems are vital for healthcare, as they can monitor and detect abnormal behaviour s in patients, helping to predict and prevent medical emergencies. Moreover, in sectors such as retail and marketing, behavior detection can offer valuable insights into consumer preferences and trends, supporting targeted advertising and product development. Lastly, these systems can contribute to improving overall efficiency and safety in various sectors by automating routine tasks and ensuring adherence to safety protocols. In essence, human behaviour detection systems serve as a proactive tool in addressing security, healthcare, business, and operational challenges in modern society.

A human behaviour detection system is essential due to its multifaceted benefits across various sectors. One critical aspect is security enhancement, where these systems can analyze and identify suspicious activities in real-time, helping to prevent crimes and ensure public safety. By monitoring behaviour al patterns, such systems can detect anomalies that might indicate potential threats or emergencies, allowing for timely intervention.

In healthcare, behaviour detection systems play a pivotal role in patient care and management. They can monitor patients' behaviour and vital signs, alerting healthcare providers to changes that

could indicate a deteriorating condition or imminent medical issue. This early detection can significantly improve response times and patient outcomes.
Moreover, behaviour detection technology is instrumental in improving efficiency and decision-making across industries. In retail, for instance, analyzing customer behaviour can optimize store layouts, product placement, and marketing strategies. In manufacturing and logistics, these systems can enhance worker safety by detecting fatigue or risky behaviour.
Overall, human behaviour detection systems are indispensable tools for security, healthcare, and business operations. They offer a proactive approach to risk mitigation, resource optimization, and public safety, making them invaluable assets in today's dynamic and interconnected world.

1.3 Objectives

Objective 1: Detection of antisocial behaviour in Hindi language posts by leveraging Multilingual BERT
Detecting antisocial behaviour using Hindi language posts presents unique challenges and opportunities in the domain of NLP and sentiment analysis. Considering the abundance of Hindi content on online platforms including social media , examining these posts can yield important insights into individuals' attitudes and behaviors. Machine learning algorithms can be programmed to recognize patterns that are indicative of antisocial behaviour, such as hate speech, harassment, or incitement of violence, within Hindi text.
To achieve accurate detection, NLP models need to account for the nuances of Hindi language, including regional dialects and colloquialisms, which can significantly influence the interpretation of text. Methods like topic modeling , entity recognition and sentiment analysis can utilized to detect problematic content and mark it for additional examination or intervention. In this project I has been using BERT for detecting antisocial behaviour using Hindi language posts.
Objective 2: Detection of prosocial behaviour in Hindi language posts by leveraging Multilingual BERT and feedback system.
Detecting prosocial behaviour through Hindi language posts presents a compelling opportunity to leverage NLP techniques for positive social impact. By analyzing Hindi content on social media and other platforms, NLP models can identify expressions of empathy, altruism, and community support within the text. This involves developing algorithms that can recognize linguistic cues indicative of prosocial behaviour, such as expressions of gratitude, willingness to help others, or advocacy for social causes.

To effectively detect prosocial behaviour in Hindi language posts, NLP systems must account for linguistic nuances, cultural context, and regional variations in expression. Techniques like sentiment analysis, semantic analysis, and context modeling can be adapted to interpret the positive intent behind Hindi text, facilitating the identification of constructive interactions and beneficial contributions to the community. In this project, I plan to utilize BERT for the purpose of identifying antisocial behaviour within Hindi language posts.

1.4 Research question
How effective is NLP in detecting various types of human behaviour, including both prosocial and antisocial behaviours, within Hindi language posts?

In assessing the effectiveness of NLP for behavior detection in Hindi language posts, it is essential to recognize the interdisciplinary nature of this research. The collaboration of experts from various fields, including linguistics, social sciences, and computer science, plays a critical role in the advancement of this technology. Linguists offer their extensive knowledge of Hindi language syntax, semantics, and cultural subtleties, which is crucial for creating NLP models that

can precisely interpret and analyze text. Social scientists contribute their understanding of human behavior and social dynamics, allowing for the detection of behavioral patterns and trends in online interactions. Meanwhile, computer scientists bring the technical skills needed to design, develop, and enhance NLP algorithms. By combining these diverse perspectives, researchers can address the unique challenges posed by the Hindi language, such as its rich morphology, diverse dialects, and context-dependent meanings. This interdisciplinary approach ensures that NLP models are not only linguistically sound but also culturally relevant, enhancing their ability to detect and interpret subtle behavioral cues accurately.

Moreover, developing robust NLP tools for behavior detection in Hindi posts can significantly impact efforts to foster positive online interactions and mitigate negative behaviors in digital environments. These instruments can be utilized to observe and analyze discussions on social media platforms, identify harmful or abusive content, and promote respectful and constructive dialogue. They can also assist in crafting intervention strategies and educational initiatives targeted at diminishing cyberbullying, hate speech, and different types of online harassment.

Ultimately, the integration of linguistic, social, and computational expertise in NLP research holds great promise for understanding and influencing human behavior in the digital age. By leveraging these interdisciplinary collaborations, we can create more effective and inclusive NLP applications that provide the diverse linguistic and cultural landscape of Hindi-speaking online communities. This, in turn, contributes to building safer and more positive online spaces for all users.

2. SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

2.1 Introduction

Purpose

The aim of this document is to detail the software requirements for developing a machine learning model to detect prosocial and antisocial behaviour in text using mBERT. This system will preprocess text data, fine-tune the mBERT model, and provide an interface for predictions.

Scope

The project aims to create a robust and accurate text classification system capable of identifying prosocial and antisocial behaviour in multilingual contexts. This will involve data preprocessing, model training, evaluation, and deployment for real-time predictions.

References

- Devlin, J., et al., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding"
- Hugging Face Transformers Documentation

Overview

This document outlines both the functional as well as non-functional requirements, covering user needs, system architecture, and technical specifications.

2.2 Product Perspective

The prosocial and antisocial behaviour detection system is a standalone application that uses the mBERT model for text classification. It will integrate with existing systems through an API for real-time predictions.

2.3 Product Functions

- Data preprocessing: Cleaning and preparing text data for training.
- Model training: Fine-tuning mBERT on labeled datasets.
- Evaluation: Assessing model performance using validation and test datasets.
- Prediction: Providing an interface for real-time text classification.
- Data storage: Storing training data, models, and predictions.

2.4 User Characteristics

- Data Scientists: Utilize the system for training and evaluating models.
- Developers: Integrate the system with other applications through APIs.
- End-users: Use the system to classify text for prosocial and antisocial behaviour .

2.5 Constraints

- The system requires a GPU for training to ensure efficient processing.
- The system must handle multilingual text efficiently.
- Data privacy must be maintained, especially with sensitive user-generated content.

Assumptions and Dependencies

- Availability of labeled datasets for prosocial and antisocial behaviour.
- The mBERT model and related libraries will be used for text processing and classification.

2.6 Functional Requirements

Data Preprocessing
- The system shall clean text data by eliminating special characters, normalizing case, and handling missing values.
- The system shall tokenize and encode text data using the mBERT tokenizer.

Model Training
- The system shall be fine-tuneing the mBERT model utilising the training dataset.
- The system shall be useing cross-entropy loss along with the AdamW optimizer during training.
- The system shall grouped the dataset to training, validation, and test subsets.

Evaluation
- The system shall assess the measurement like accuracy, precision, recall, F1-score.
- The system shall provide validation and test loss after each epoch.

Prediction
- The system shall provide an API for real-time text classification.
- The system shall return whether a given text is prosocial or antisocial.

Data Storage
- The system shall store training data, model checkpoints, and prediction results.

2.7 Non-functional Requirements

Performance
- The system shall practice text , provide predictions in real-time (under 1 second per request).

Scalability
- The system shall manage a large volume of requests efficiently.

Reliability
- The system shall have high availability, with minimal downtime.

Security
- The system shall guarantee the privacy of data and secure communication through encryption.

Usability
- The system should feature an easy-to-use interface for accessing and interpreting predictions.

2.8 System Architecture

High-Level Architecture

- Data Ingestion: Collect and preprocess text data.
- Model Training: Fine-tune mBERT on preprocessed data.

- Evaluation: Validate model performance.
- Prediction API: Real-time text classification endpoint.
- Data Storage: Database for storing data and model artifacts.

Detailed Design

- Preprocessing Module: Handles data cleaning and tokenization.
- Training Module: Manages model fine-tuning and optimization.
- Evaluation Module: Computes evaluation metrics.
- Prediction Module: Provides an API for text classification.
- Storage Module: Manages persistent storage of data and models.

2.9 User Interface

User Interface Requirements

- The system shall provide visualization of evaluation metrics and model performance.

Other Requirements

Environmental Requirements

- The system shall operate on cloud infrastructure for scalability and reliability.
- The system will accommodate various languages for text input and classification.

Documentation Requirements

- The system shall include comprehensive documentation for users and developers.
- The system shall provide API documentation for integration purposes.

Legal Requirements

- The system shall comply with data protection regulations, including GDPR.

This SRS document outlines the detailed requirements for developing a prosocial and antisocial behaviour detection system using mBERT. It covers system architecture, functional and non-functional requirements, and user interface design, providing a complete guide for development and implementation.

2.10 Hardware Requirements

To successfully implement and run the prosocial and antisocial behaviour detection system using mBERT, the following hardware components are required:
Development and Training Environment

Local Development Machine (Optional)

- Processor: Intel Core i7 or equivalent
- Storage: At least 500 GB of free space with SSD
- RAM: 16 GB or higher
- GPU: NVIDIA GPU with at least 8 GB VRAM for initial development and small-scale experiments
- Operating System: Linux (Ubuntu 18.04 or later), Windows 10, or macOS

Cloud Computing (Recommended for Training and Production)

- Processor: Multi-core CPUs with high performance (e.g., AWS EC2 C5 instances, Google Cloud Compute Engine N1 instances)
- RAM: 64 GB or higher
- Storage: At least 1 TB of storage with SSD for datasets and model checkpoints
- GPU: NVIDIA Tesla V100, A100, or equivalent with at least 16 GB VRAM for efficient training and inference
- Operating System: Linux-based OS (Ubuntu 18.04 or later)

3. FEASIBILITY STUDY

A feasibility study assesses the feasibility and practicality of a proposed project. In the case of developing a system for detecting prosocial and antisocial behaviour using mBERT, several aspects need consideration:

3.1 Technical Feasibility

Availability of Technology: The necessary technology, including mBERT model and NLP libraries, is readily available and well-documented. Therefore, the technical implementation is feasible.

Data Availability: Availability of labeled datasets for prosocial and antisocial behaviour s is crucial. If suitable datasets are accessible, the system's technical feasibility is high. Otherwise, data collection efforts may be required, affecting project timelines and costs.

3.2 Economic Feasibility

Cost of Development: Development costs include hardware infrastructure, software licenses, and personnel expenses. Cloud computing resources may incur ongoing operational costs, but they offer scalability and flexibility.

Return on Investment (ROI): The system's ROI depends on its application and market demand. If deployed in industries such as social media monitoring, mental health support, or content moderation, the potential for ROI is significant.

3.3 Operational Feasibility

Scalability: The system needs to be capable of managing different volumes of text data and user requests effectively. Cloud-based deployment allows for easy scalability by provisioning additional resources as needed.

Integration: Integration with existing systems or platforms, such as social media platforms or mental health applications, may be required. Compatibility and interoperability with these systems should be ensured.

3.4 Legal and Ethical Feasibility

Data Privacy and Security: Adherence to data privacy laws (such as GDPR) is essential, especially when processing user-generated content. Ensuring data security and user anonymity are critical considerations.

Ethical Implications: Detecting sensitive behaviour s such as antisocial tendencies requires careful handling to avoid stigmatization or privacy violations. Ethical guidelines and mechanisms for bias detection and mitigation should be incorporated into the system.

3.5 Schedule Feasibility

Project Timeline: The project timeline depends on factors such as data availability, development complexity, and testing requirements. Agile development methodologies can help manage project timelines effectively.

Milestones and Deliverables: Clearly defined milestones and deliverables ensure project

progress tracking and timely completion. Regular review meetings and agile sprints facilitate iterative development and adaptation to changing requirements.

Overall, developing a system for detecting prosocial and antisocial behaviour using mBERT is technically feasible, provided suitable datasets are available. Economic feasibility depends on development costs and potential ROI in relevant industries. Operational feasibility requires ensuring scalability, integration, and adherence to legal and ethical standards. With careful planning and execution, the proposed system holds promise in addressing social and mental health challenges in digital environments.

3.6 Existing systems:

Existing systems for detecting prosocial and antisocial behaviour in text leverage various NLP techniques along with machine learning models. While each system has its unique features and approaches, they share common goals of enhancing online safety, fostering positive interactions, and mitigating harmful behaviour. Here are some examples:

Social Media Monitoring Tools:

Advantage: These tools offer comprehensive monitoring capabilities, allowing businesses to monitor brand sentiment, spot emerging trends, and interact with customers instantly.

Disadvantage: They may struggle with context and nuance, leading to misinterpretation of sarcasm, humor, or culturally specific expressions, resulting in inaccurate sentiment analysis.

Content Moderation Systems:

Advantage: Content moderation systems can proficiently process vast amounts of user-generated content, helping platforms maintain a safer and more comprehensive online environment.

Disadvantage: They may exhibit biases, as Machine learning algorithms, when trained on historical data, have the potential to unintentionally sustain existing biases or fail to recognize subtle forms of harmful behaviour .

Mental Health Support Platforms:

Advantage: These platforms offer accessible and convenient mental health support, reaching individuals who may not have access to traditional therapy or counseling services.

Disadvantage: They may face challenges in accurately detecting nuanced expressions of distress or mental health issues in text, potentially leading to missed opportunities for intervention or support.

Academic Research Projects:

Advantage: Academic research projects contribute to the progression of knowledge and comprehension within the realm of NLP and behavior detection, driving innovation and informing the development of more robust detection algorithms.

Disadvantage: Research findings may not always translate effectively into practical applications, and the pace of progress may be slower compared to commercial systems due to resource constraints and academic publication timelines.

Community-driven Initiatives:

Advantage: Community-driven initiatives promote user empowerment and foster a sense of ownership and responsibility among online communities, encouraging positive behaviour and self-regulation.

Disadvantage: They may rely heavily on volunteer moderators, leading to inconsistencies in enforcement and moderation practices across different communities. Additionally, they may struggle to address systemic issues or coordinate responses to complex challenges such as coordinated harassment campaigns.

The proposed system for detecting prosocial and antisocial behaviour using mBERT offers several advantages over existing systems:

3.7 Advantages of the propose system:

Language Agnostic: mBERT is a multilingual model capable of understanding and processing text in various languages, making the system applicable to diverse linguistic contexts.

Fine-tuning Capability: By fine-tuning the mBERT model on labeled datasets specific to prosocial and antisocial behaviour s, the system can adapt to the nuances of different behaviour types, improving accuracy and performance.

Real-time Prediction: The system provides real-time text classification, enabling quick identification of prosocial and antisocial behaviour in online interactions, social media posts, or user-generated content.

Customization and Scalability: The system's modular architecture allows for easy customization and scalability. It can be tailored to specific use cases and integrated with existing platforms or applications through APIs.

3.8 Handling Disadvantages:

Contextual Understanding: To address the challenge of context and nuance faced by existing systems, the proposed system can utilize contextual embeddings and attention mechanisms in mBERT. These techniques empower the model to capture subtle linguistic cues and context-specific meanings, enhancing its understanding of text.

Bias Mitigation: To mitigate biases inherent in machine learning models, the proposed system can employ bias detection and mitigation techniques during model training and evaluation. This includes diversifying training datasets, incorporating fairness metrics, and post-processing techniques to address biases in predictions.

Detection of Nuanced Expressions: To improve the detection of nuanced expressions of behaviour, the system can incorporate additional features such as sentiment analysis, emotion recognition, and linguistic style analysis. By combining multiple signals and features, the system can better capture the complexity of human communication and behaviour.

Community Engagement: To promote community engagement and user empowerment, the system can include features for user feedback, reporting mechanisms, and community moderation tools. This encourages users to actively participate in shaping online behaviour norms and fosters a sense of collective responsibility for maintaining a positive online environment.

By leveraging the capabilities of mBERT and incorporating advanced techniques for bias mitigation, context understanding, and user engagement, The proposed system seeks to address the limitations of current systems and offer a strong solution for detecting prosocial and antisocial behaviour in text.

4. LITERATURE SURVEY

To ensure the projects's relevance, extensive reference and consultation with various other papers and journals were undertaken.

F. Shannaq et. al [9]: This study focuses on developing an innovative predictive system with two-stage approach for optimization to effectively categorize offensive and non-offensive text. In the first stage, it refines a pre-trained embedding for words through multiple training epochs with the provided dataset, enhancing the embeddings of vocabulary items specific to the dataset and integrating them with existing embeddings. Subsequently, a hybrid approach is utilized in the second stage, two classifiers such as XGBoost and SVM—in conjunction with a genetic algorithm to reduce the classifier limitations and optimize hyperparameter to get their values. The evaluation was conducted on the Arabic Cyberbullying Corpus (ArCybC), which comprises tweets from diverse fields such as news, sports ,gaming, and celebrities, considered into themes related to sexual, cultural, intelligence, and appearance topics. The findings demonstrate significant effectiveness, particularly with SVM employing the Aravec SkipGram model for word embedding, attaining an impressive accuracy rate equalto 88.2% , F1-score equalto 87.8%. This underscores the system's robust capability in accurately identifying and categorizing offensive content within varied Arabic language tweets.

Viera Maslej-Kreš ˇ náková et. al [10]: This research investigates the efficacy of Easy Data Augmentation (EDA) techniques, which involve simple text transformations to generate synthetic samples, for identifying antisocial behavior. Specifically, the study focuses on mitigating class imbalance issues in two specific tasks: classifying fake news and detecting toxic comments. We train convolutional neural network (CNN) classifiers on both original datasets and those extended with EDA, then assess their performance.

The findings indicate that the effectiveness of EDA techniques is influenced by the task and data characteristics. For the extended toxic comments dataset, the incorporation of a subset of EDA methods resulted in only minor improvement, with a slight 0.01 increase in the F1 score. These methods were less effective for processing informal language.

In contrast, the fake news dataset showed a more notable enhancement in performance with EDA. F1 score increased by 0.1, especially in predicting the class of minority, in this study F1 score improved from 0.67 to 0.86. This highlights the potential of EDA to significantly improve performance in specific contexts, such as boosting classification accuracy for certain categories like fake news. However, the effectiveness of EDA can differ based on the dataset's features and the specific task at hand.

Ojasv Kamal et al [11]: This paper discussed a transfer learning method designed to classify social media posts in the Hindi Devanagari script, gathered from platforms like Twitter as well as Facebook, into either Hostile or Non-Hostile categories. The classification of Hostile posts extends to identifying specific attributes associated with Hateful, Fake, Defamation, and Offensive content. The methodology utilizes attention-based pre-trained models for Hindi data. The primary objective is to classify posts into Hostile class group and Non-Hostile groups, followed by incorporating features for the sub-task classifications. The aim is to develop a robust and consistent model without resorting to ensemble methods or intricate pre-processing techniques.

Vashistha, N et al [12]: In this research, the scholars examined and amalgamated six openly accessible datasets to construct a unified dataset categorized consistently into three groups: hateful ,abusive, or neutral. They established a foundational model and enhanced its effectiveness

through various optimization methods. The attained performance level prompted the creation of a tool capable of swiftly evaluating web pages almost in real-time using an efficient metric. Following feedback, the model underwent retraining.

Moreover, the team demonstrated the competitive performance of a multilingual model adept at handling both English and Hindi. They presented impressive outcomes, achieving a 95% accuracy rate that surpassed or equaled the performance of numerous monolingual models in the field.

Shervin Malmasi et al[23]: In this research, the team tackles the challenge of distinctive between general blasphemy and hate speech in social media content. They utilize a newly created dataset specifically annotated for this purpose. Employing supervised classification methods, the team incorporates a diverse feature set includes representations of skip-grams, clustering-based word representations.

Their approach involves the use of both individual classifiers and ensemble classifiers like stacked generalization. The finest outcome achieved for this three-class classification task is an accuracy rate of 80%. This methodology showcases the effective application of advanced classification techniques to discern varying levels of offensive language and hate speech in online content.

Vikas Kumar Jha et.al [6] : The authors emphasize the dual nature of social media, which serves as a platform for sharing thoughts, experiences, and emotions but also presents challenges when interactions deteriorate into spaces for abusive language and offensive remarks. Such language, which may include derogatory terms, is often used to express disdain, disagreement, or humor but can also perpetuate racism and sexism. Hindi, being the third most spoken language globally, exhibits significant diversity due to regional influences and linguistic variations. While "Hinglish" which is written in the Snglish script is commonly used online, there is a noticeable increase in native Hindi speakers opting for the Devanagari script.

This study presents a novel model designed to distinguish between offensive group text and non-offensive group text utilizing a fastText-based approach. The model which is proposed in this paper effectively categorized text taken from the Devanagari Hindi Offensive Tweets (DHOT) corpus, achieving an impressive accuracy rate of 92.2% on a desktop-class machine. By employing a grid-search method, the study fine-tuned hyperparameters throughout the fastText model iterations, offering important insights into the model's accuracy and precision. This research marks a significant step forward in establishing a cutting-edge classification system for offensive text in Hindi, leveraging fastText models.

Jisu Kim, et.al [35]: This study highlights the significance of considering platform design, structure, and functionalities in encouraging civility and fostering pro-social behaviors among users. It emphasizes how the features of a platform can influence user behavior and the dynamics of interaction, stressing the importance of thoughtful platform design in nurturing positive online engagements. Through a detailed quantitative analysis of observations on Nextdoor, the research found that platform layout significantly impacts the level of civility in discussions. Comments made within specific groups showed higher civility levels and were less frequently flagged for moderator intervention compared to those posted on the general neighborhood feed. Additionally, groups that introduced guidelines to new members experienced fewer instances of inappropriate content reports and exhibited a more decently conscientious tone in observations, contrasting with groups where guidelines were not introduced to new members.

Coe et.al [36] : This research scrutinized a 3-week survey of articles and comments published on a local newspaper's website, covering more than 300 articles and 6,400 comments. The results of the content analysis reveal that occurrences of incivility are common and are linked to specific contextual elements, including the article's topic and the sources cited.Moreover, the study

discovered that, contrary to popular belief, regular commenters tend to exhibit more civility compared to occasional commenters. Furthermore, uncivil commenters display a similar inclination as civil commenters in using evidence to support their arguments.

Sl No.	Paper/authors/year	Language	Dataset	Category	Feature	Algorithm	Highest Performance
	Kamal et al /2022	Hindi	Twitter, Facebook	Hateful, Fake, Defamation, and Offensive	-	shared BERT	F1 Score= 0.95
	Vashistha, N et al .; 2021	English ,Hindi, Hindi code mix	own resource of hate speech sequences	abusive, hateful or neither	BoW, TFIDF	logistic regression	F1= 0.96 accu =0.95
	Mohit Bhardwaj† et al /2020	Hindi, English, Roman Hindi	Manually annotate∼8200 onlineposts.	Hateful, Fake, Defamation, and Offensive or neither	m-BERT	LR,SVM,RF	SVM coarse grain F1=84.11
	Aditya Bohra et al /2018	English Hindi code-mixed	Twitter	Normal Speech Hate Speech	Character N-Grams, WordN-Grams(W) Punctuations, NegationWords	Random forest	Accuracy= 71.7%
	Vikash Kumar jha et al	Hinlish (Roman hindi)	Twitter	Offensive, non-offensive	-	FastText	Accuracy = 92 %
	Prabhat Agarwal et al /2017	English and Romanized Hindi	Twitter	swear words, non-swear words	Consecutive Repeated Character Removal, Phonetic Match swear words	-	Precision=0. 73 and recall = 0.96
	Kakwani, D.,et al	11indian	News crawls	News category	-	FastText	Accuracy =77.39 %

	2020	languages and Indian English		classification			
	Mathur, P.; et al /2018	Hinhlish	Tweet	Abusive, Hate indicating, non-offensive	Bow, TF-IDF	Multi-Input Multi-Channel TransferLearning	F1 score =0.723
	K Sreelakshmi et al./ 2019	Hindi-English code-mixed	Facebook	Hate speech, non- hate speech	CBOW and skip-gram. Doc2vec	SVM-RBF	Accuracy=85.81 %
	Satyajit et.al/2018	Hindi-English	Tweeter	Hate speech, non- hate speech	word2vec	CNN-1D	Accuracy= 82.62 %
	Kumar, S., & Singh, T. D. (2022)	Hindi	Hindi Fake and True Dataset	Fake, True	-	Logistic regression,Naive Bayes, LSTM	Accuracy = 92.36%

Table 1

5. BACKGROUND

NLP encompasses a variety of techniques and methods used The ability to interpret, analyze, and comprehend human language is essential by computers. It encompasses the creation and implementation of algorithms and models for tasks such as text classification of text,NER sentiment analysis of text, machine translation, and speech recognition. NLP techniques include tokenization of sentences, part-of-speech(POS) tagging of terms named entity recognition(NER) is recognizing entities like names, places, or locations, and syntactic parsing is evaluating the grammatical construction of sentences. These techniques are vital for extracting understandings from huge capacities of text data and aiding applications across various industries, from improving customer service to automating information retrieval and analysis. NLP continues to advance rapidly, driving innovation in how computers interact with and understand human language.

5.1 Preprocessing

Noise Removal: Removing noise from Hindi text involves eliminating unwanted text such as HTML tags in a sentences , punctuation marks in that sentence, special characters and non-textual content to clean the data and improve its quality for NLP tasks. In Hindi text, noise can include extra spaces, newline characters, numerical digits, symbols, or diacritics that do not contribute to the linguistic meaning. Techniques for noise removal typically involve using regular expressions or specific string manipulation methods to filter out or replace unwanted characters and patterns. For example, one can use Python's `re` library to remove non-alphabetical characters and normalize the text by replacing multiple spaces with a single space. Noise removal is a crucial preprocessing phase in Hindi NLP. To enhance the precision and efficiency of future text analysis, categorization, or information retrieval endeavors.

Example Hindi text with noise

Original Hindi Text:

यह है एक उदाहरण 123! कि हिन्दी में noise कैसे हटाए जाते हैं।

Cleaned Hindi Text after Noise Removal:

यह है एक उदाहरण कि हिन्दी में noise कैसे हटाए जाते हैं।

Normalization

Normalization in Hindi text processing includes transforming raw data into a standardized and consistent format to enhance its readability and suitability for NLP applications. This method addresses various aspects of text normalization specific to Hindi, such as standardizing characters, handling numerals, and managing diacritics. Common techniques of normalization include converting all text to a uniform script (e.g., Devanagari), replacing numeric representations with words (e.g., converting "१२३" to "तीन सौ इक्कीस"), and removing diacritics to simplify text representation. Proper normalization of Hindi text is essential for refining the accuracy and efficiency of succeeding NLP tasks, as tokenization, lemmatization, and sentiment analysis, by reducing data redundancy and aligning text representations with linguistic conventions[30].

Stopword Removal: Stopword removal is an essential preprocessing step in Hindi text processing aimed at filtering out commonly utilized terms that lack specificity,substantial semantic meaning and may introduce noise in NLP tasks. Stopwords in Hindi can include frequently occurring words such as "और", "के", "का", "है", "यह", "वह", etc., which are necessary for grammatical structure but often do not contribute to the core content or context of the text. To perform stopword removal in Hindi, a list of stopwords specific to the language is typically compiled. This list is then used to filter out stopwords from the input. Tokenization of the text into specific words or tokens is followed by the elimination of tokens that match any word in the

stopwords list. The result is a cleaner version of the text that retains only the expressive words, improving the efficiency and accuracy of subsequent NLP tasks similar to text classification, sentiment analysis, and information retrieval. Proper handling of stopwords is crucial in Hindi NLP to confirm that the analysis focuses on relevant text and avoids processing unnecessary linguistic elements that can impact performance and interpretation.

Stemming: Hindi stemming is a linguistic process in NLP to Moreover, online systems provide personalized experiences and tailor suggestions based on user preferences, enhancing user fulfilment and engagement [9]. This process helps in normalizing different types of words to their common base, which can aid in tasks like text mining , information retrieval, , and search engine optimization. In Hindi, stemming is particularly important due to the language's rich morphology and complex word formation rules. Common stemming techniques for Hindi involve applying algorithms that understand the structure of Hindi words and can accurately identify and remove suffixes like -ो, -ीयों, -एँ, etc., to derive the root form. Effective Hindi stemming can improve the accuracy and efficiency of NLP applications by reducing word variations and treating related words as equivalent, thus enhancing the overall processing of Hindi text data.

Lemmatization

Lemmatization in Hindi text processing includes dropping words to their base forms or as mentioned in dictionary, known as lemmas, to normalize variations and improve the accuracy of NLP tasks[43]. In Hindi, lemmatization is particularly vital due to the language's complex morphology and extensive inflections. The goal of lemmatization is to convert different forms of a word into a common base form, which helps in reducing redundancy and ensuring consistency in data analysis. Unlike stemming, which often involves heuristic rules to chop off affixes, lemmatization uses linguistic rules and dictionaries to derive the canonical form of words. For ex., lemmatization of the the word "खेलती" (playing, feminine) would result in "खेलना" (to play) as the lemma. Lemmatization is beneficial in Hindi NLP for tasks like text normalization, machine translation, where understanding the underlying linguistic structure is crucial for accurate analysis and interpretation of text data[44].

Tokenization

The aim of tokenization is to facilitate computational analysis of Hindi text by breaking it down into discrete units that can be processed effectively. This step is essential for tasks like part-of-speech tagging of terms or words, named entity recognition of words, and sentiment analysis of text in Hindi language processing. Different tokenization techniques exist for Hindi, considering the unique characteristics of the language, such as compound words and postpositions. Effective tokenization is critical for accurately representing the linguistic structure and semantics of Hindi text in NLP applications.

Tokenization of Hindi text involves several distinct methods tailored to the specific linguistic features of the language. One well known approach is word-based tokenization, in which the text is tokenized into individual words or terms based on spaces or delimiters. Another method is morphological tokenization, which breaks down words into their constituent morphemes, including stems and affixes, reflecting the rich morphology of Hindi. Character-based tokenization, on the other hand, treats each character as a separate token, which can be useful for certain NLP tasks like transliteration and handwriting recognition. Additionally, subword tokenization techniques like WordPiece ,byte-pair encoding are employed for out-of-vocabulary words and optimize vocabulary size in machine learning models. Each type of tokenization in Hindi serves specific purposes and can be selected based the NLP task at hand, ensuring effective processing and analysis of Hindi text data.

5.2 Name Entity Recognition (NER)

NER in Hindi text processing involves identifying the terms and categorizing those into named entities such as names of persons, organizations, locations, dates, and other specific entities within the text. NER is a key task in NLP that helps extract important information from unstructured text data. In Hindi, NER faces unique features due to the language's rich morphology and context-dependent nature [55].
To perform NER on Hindi text, several approaches can be employed:

Rule-based NER

NER for Hindi text involves developing linguistic rules to identify and categorize named entities of text within the sentences based on syntactic and semantic indications specific to the Hindi language[57]. This approach relies on defining rules to recognize named entities like person's name , organization's name, locations, dates, and other entities based on characteristic features such as capitalization, affixes such as prefixes, suffixes, and contextual information. For example, to identify names of persons in Hindi text, rules may be defined to recognize sequences of words with specific honorifics (e.g., "श्रीमान", "श्रीमती") followed by proper nouns. Similarly, rules can be designed to identify locations based on geographical names, or dates based on numerical patterns and contextual phrases. Rule-based NER offers transparency and control over the recognition process, allowing customization and adaptation to domain-specific requirements. However, it may require consistent refinement and maintenance as language usage evolves and new entities emerge. Despite its limitations, rule-based NER serves as a foundational approach for named entity extraction in Hindi NLP, complementing other techniques like machine learning and dictionary based methods.

Dictionary-based NER

Dictionary-based NER for Hindi text involves leveraging curated dictionaries or lists of known named entities such as names of persons, organizations, locations, and other specific terms to identify and classify entities within the text. This approach relies on matching tokens in the text against entries in the pre-defined dictionaries to recognize entities based on exact or approximate string matching. For example, a dictionary-based NER system for Hindi may contain entries of common Hindi names, cities, states, and well-known organizations. During processing, tokens from the input text are compared against these dictionaries, and if a match is found, the corresponding entity label is assigned. Dictionary-based NER is relatively straightforward to implement and can be effective for recognizing commonly occurring entities in specific domains or contexts. However, it may suffer from limitations such as the need for comprehensive and up-to-date dictionaries, difficulty in handling variations and misspellings, and limited coverage of less common or emerging entities. Despite these challenges, dictionary-based NER remains a useful approach for named entity extraction in Hindi NLP tasks, especially when combined with other techniques like rule-based methods or machine learning-based models to enhance performance and coverage.

Machine Learning-based NER

This approach leverages annotated datasets of Hindi text where entities are labeled, and features are extracted to train supervised learning models. Commonly used algorithms for NER include Conditional Random Fields (CRF), deep learning and Support Vector Machines (SVM) architectures like Bidirectional LSTMs (Long Short-Term Memory) with CRF layers [59]. Machine learning models are learned patterns and relationships through training on annotated Hindi corpora to between words that signify named entities. These models capture contextual information and dependencies among words to predict entity labels for each token in the text sequence.

Pretrained Language Models:
Pretrained Language Models (PLMs) have revolutionized Named Entity Recognition (NER) [56] for Hindi text by enabling advanced contextual understanding and accurate entity prediction. PLMs like Bidirectional Encoder Representations from Transformers and its different types offer powerful capabilities for NER tasks in Hindi. These models are pre-trained for NER on large-dataset and can be fine-tuned on any field-specific data to improve performance on NER.

5.3 Sentiment analysis

Sentiment analysis for Hindi text involves the process of discovering the emotion evident in a section of Hindi text, whether it is positive, negative, or neutral. This task is crucial for grasping public opinion, analyzing customer feedback, or gauging sentiment on social media. in the Hindi language domain. Several techniques can be applied to perform sentiment analysis on Hindi text [54].There are several methods for executing sentiment analysis on Hindi text, each leveraging different techniques and methodologies. Here are the main types of approaches used for sentiment analysis of Hindi text[53]:

Lexicon-based Approach
This approach involves using sentiment lexicons in a text segment or dictionaries that allocate sentiment scores to Hindi words. Each word is associated with a polarity (positive, negative, neutral) based on its meaning. Sentiment scores are aggregated to compute the overall of a word in a sentence. Lexicon-based methods are straightforward but may not be able to determine context-specific sentiments effectively.

Machine Learning-based Approach:
Machine learning algorithms, for example Support Vector Machines (SVM) for sentiment analysis , Naive Bayes for sentiment analysis, Random Forests for sentiment analysis, or deep learning models like Recurrent Neural Networks (RNNs) for sentiment analysis [42] and Convolutional Neural Networks (CNNs), for sentiment analysis can be used. These models are developed using labeled Hindi text data in a sentence, where each sample is tagged with sentiment labels such as positive label, negative label , or neutral label . The trained model learns to predict the sentiment of unsen text based on learned patterns and features extracted from the data. Transfer learning techniques can be applied using pretrained language models like BERT or its variants trained on large-scale Hindi text corpora. These models are fine-tuned on sentiment analysis tasks using labeled data, enabling them to capture complex linguistic patterns and context-specific sentiment cues in Hindi text.

Ensemble Methods:
This type of SA methods enhance the accuracy and robustness of sentiment analysis by integrating multiple models or classifiers. By combining predictions from various models, these methods can reduce individual model biases and improve overall performance [52].

5.4 Text summarization

Text summarization in Hindi involves the process of shrinking a piece of text while retaining its important information and sense.
To conduct summarization on Hindi text, various methods can be utilized:

Classification based
Classification-based text summarization for Hindi text involves a method where the text is categorized into different classes or topics, and then a summary is generated based on the most representative or important content within each category. This approach combines text

classification and summarization techniques to efficiently shrink large volumes of Hindi text into brief summaries tailored to specific topics or themes. The process typically begins with preprocessing steps such as tokenization, stopword removal, and feature extraction. Next, a classification model is the model is trained using labeled Hindi text data to forecast the category or topic of each input document. Once the classification model is trained and validated, text summarization is performed separately for each category using extractive or abstractive methods to generate informative and coherent summaries. Classification-based text summarization is valuable for organizing and summarizing diverse Hindi text datasets, enabling efficient information retrieval and content understanding across various domains and applications. This approach can be adapted and refined using Utilizing machine learning methods along with specialized training data for specific domains. to optimize summarization performance for Hindi language processing tasks.

Clustering based

Clustering-based text summarization for Hindi text it includes organizing similar documents or sentences into clusters of terms according to their into clusters depending on their semantic or contextual similarity, and then generating summaries by selecting representative sentences or key phrases from each cluster . This approach leverages unsupervised learning techniques to organize large volumes of Hindi text data into coherent clusters, which helps identify common themes or topics within the text corpus.

To implement clustering-based text summarization for Hindi, the process typically begins with preprocessing steps including tokenization, stopword removal, and vectorization of text data. Next, clustering algorithms such as K-means clustering and, hierarchical clustering for text data, or density-based clustering for text data are applied to collection of similar documents or sentences organized based on their feature representations. Once the clustering is done, summary generation involves selecting sentences or phrases that best represent each cluster, ensuring coverage of important information while minimizing redundancy.

Graph based

Graph-based text summarization for Hindi involves representing the Represent the text as a graph where nodes correspond to sentences or phrases, and edges denote the relationships between them based on similarity metrics like cosine similarity or semantic similarity [62]. This approach leverages graph algorithms to identify important nodes (sentences) that serve as key points in the text, facilitating the extraction of essential information for generating a summary.

To perform graph-based text summarization for Hindi, the process typically begins with preprocessing steps including tokenization, stopword removal, and sentence segmentation. Next, sentence embeddings are computed using techniques like word embeddings Word2Vec or pretrained language models as BERT to capture semantic relationships between sentences. These embeddings are then used to construct a similarity matrix in the dociment, where each entry represents the similarity score between pairs of sentences. Once the similarity matrix is constructed, graph algorithms such as PageRank or TextRank are applied to identify important Sentences are chosen to create the summary based on their prominence within the graph. The highest-ranked sentences are then included, ensuring that the generated summary captures the key information and main ideas from the original Hindi text.

Optimization based

Optimization-based text summarization for Hindi [65] involves using mathematical optimization techniques to choose a subset of sentences or phrases from the original text that capture the key information while meeting specific constraints. This approach formulates the summarization task

as an optimization problem in Natural Language Processing where the goal is to maximize the informativeness and quality of the summary based on defined criteria, such as sentence importance and coverage of key concepts, while considering constraints like summary length or diversity of content. By applying optimization algorithms, such as Integer Linear Programming (ILP) or greedy algorithms, the system efficiently identifies the most relevant content for generating concise and informative summaries tailored to Hindi text data.

Fuzzy based

Fuzzy logic-based text summarization for Hindi involves using fuzzy sets and reasoning to evaluate the relevance of sentences in the original text for summary inclusion. This approach handles linguistic ambiguity and uncertainty by fuzzifying key linguistic features like word frequencies and semantic relationships[66]. Fuzzy logic rules are defined to assess sentence importance based on informativeness, coherence, and diversity. By leveraging fuzzy logic principles, this method accommodates the nuances of Hindi language semantics, producing concise and contextually relevant summaries that capture the essential content of the text.

Neural Network based

Neural network-based technique of text summarization for Hindi involves using deep learning models like BERT or GPT, adapted for Hindi language[60], to generate concise summaries from large text datasets. This approach converts Hindi text into numerical representations using tokenization and embeddings, then trains a model to predict summary sequences based [60] on input text. The neural network learns contextual relationships and semantic meanings, producing abstractive summaries by focusing on relevant content with attention mechanisms. This method enables automatic learning and adaptation to Hindi language nuances, providing effective summarization for various NLP tasks in Hindi text processing.

5.5 Text classification

Text classification for Hindi language text involves categorizing Hindi text classifying content into predefined categories. This task is crucial for various NLP applications such as sentiment analysis, topic classification, spam detection, and more. Here are different approaches and techniques commonly used of Hindi language text for text classification:

Traditional Machine Learning Approaches: Old-style machine learning algorithms for NLP such as Support Vector Machines (SVM) [50], Logistic Regression, Naive Bayes for text classification, Decision Trees, and Random Forests for text classification can be used for text classification of Hindi text. These algorithms rely on features extracted from the text, such as bag-of-words representations, Term Frequency-Inverse Document Frequency (TF-IDF) for text classification vectors, or word embeddings for feature detection, to train classifiers that can predict the category of unseen text.

Deep Learning Approaches: Deep learning models, particularly neural networks, have shown great success in text classification tasks for Hindi text. Architectures like Transformer-based models (e.g., BERT, GPT) , Long Short-Term Memory (LSTM) , Recurrent Neural Networks (RNNs), Convolutional Neural Networks (CNNs) can be used for text classification [48][50]. These models develop layered representations of text data, identifying intricate patterns and relationships, which enhances classification accuracy.

Transfer Learning with Pretrained Language Models: Transfer learning techniques leveraging pretrained language models for text classification like BERT or multilingual models trained on large-scale text dataset can be applied to text classification tasks for Hindi text[52][59]. Fine-tuning done on these models on labeled data specific to the classification task, enabling them to learn contextual representations and achieve maximum efficiency in Hindi text classification.

Ensemble Methods: Ensemble methods group together multiple classifiers or models to enhance the global accuracy and robustness of text classification. By aggregating predictions from different models, ensemble methods can mitigate biases and variance inherent in individual models, leading to enhanced performance on classification tasks [51].

Feature Engineering and Selection: Feature engineering techniques play a vital role in text classification. Methods like TF-IDF, word embeddings (e.g., Word2Vec, FastText), n-gram representations, and syntactic or semantic features can be used to represent Hindi text effectively for classification tasks. Feature selection techniques aid to identify the most informative features that contribute to classification performance.

5.6 Machine translation

Machine translation for Hindi language text involves using automated systems to translate text from Hindi to another language or vice versa. This technology plays a crucial role in breaking down and overcoming language barriers and enabling communication among diverse linguistic groups. Here are the key components and approaches used in machine translation for Hindi:

Statistical Machine Translation (SMT)

SMT is one of the traditional approaches used for translating Hindi text. SMT models rely on statistical models that learn from large bilingual corpora.From these dataset it learns translation patterns. These models break down the translation process into several steps, including language modeling, alignment modeling, and decoding to generate the translated output. Phrase-based SMT and word-based SMT are common techniques used in translating Hindi to other languages.

Neural Machine Translation (NMT)

NMT represents the modern approach in machine translation, including translating Hindi text. NMT models leverage deep neural networks, such as attention mechanisms followed after sequence-to-sequence embedding[47], to learn direct mappings between input (Hindi) and output (target language) sequences. Models like Transformer architecture such as Google's pretrained BERT, and OpenAI's pre trained GPT have significantly improved translation quality by capturing long-range dependencies and context in translations.

Phrase-based Translation

In phrase-based translation, the input Hindi text is segmented into phrases, and each phrase is translated into the target language based on translation probabilities learned from bilingual data. This approach focuses on translating fixed-length segments of text, resulting in relatively fluent translations.

Rule-based Translation

Rule-based translation systems use linguistic rules and grammatical patterns to translate Hindi text into another language. These systems rely on predefined dictionaries, grammar rules, and syntactic structures to produce translations. While rule-based translation can be precise for specific domains, it may struggle with handling complex linguistic variations.

Hybrid Approaches

Hybrid machine translation systems combine different approaches, such as combining statistical and neural techniques or integrating rule-based components with statistical models, to achieve more accurate and robust translations. These hybrid systems leverage the strengths of each approach to mitigate weaknesses and enhance overall translation quality.

Data Augmentation and Fine-tuning

Machine translation systems for Hindi benefit from data augmentation techniques, where

synthetic or parallel data is generated to enhance the diversity and quality of training data. Fine-tuning pretrained language models on Hindi translation tasks also improves translation quality by adapting models to specific linguistic nuances and vocabulary.

5.7 Type of antisocial content

The automatic identification of abusive text content in NLP is a complex task because there are differing opinions on what constitutes abusive content. Additionally, content that may be considered hateful by some individuals may not be viewed the same way by others, depending on their specific definitions. there are many types of abusive contents available in online social media, below are the major categories mention in Ref [15].

Abusive and Offensive language

Abusive and offensive language online presents a significant and complex challenge in digital spaces, impacting the well-being and inclusivity of online communities. This type of communication involves language that is harmful, derogatory, or intended to harm or belittle others. It spans a spectrum, from direct personal attacks to more subtle forms of discrimination based on characteristics like race of a person , gender, ethnicity of a group, or other protected attributes. Offensive language often aims to provoke, insult, or cause emotional distress. Exposure to abusive and offensive language can have profound consequences, including emotional harm, increased anxiety, and the creation of a hostile online environment. Addressing this issue requires a multi-faceted approach, which includes implementing strong content moderation policies, educating users on respectful communication, and leveraging automated tools to detect and filter harmful content.

Creating a digital culture that values constructive dialogue and discourages offensive language is essential for cultivating a safe and inclusive online community. Both platforms and users share the responsibility of actively addressing and mitigating the impact of abusive and offensive language to ensure a more positive and respectful online experience for everyone involved.

Cyberbullying

Online harassment, known as cyberbullying, is a troubling trend that utilizes digital communication platforms to deliberately bully, intimidate, or harm others. It goes beyond name-calling to encompass threats, relentless harassment, and even impersonation, revealing private information without consent ("doxing"), and outing someone's identity. The emotional The effects of cyberbullying are substantial, leading to distress, anxiety, and depression in those targeted. Technology amplifies the issue by making bullying persistent and far-reaching. Cyberbullies may exclude targets from online groups or spread their secrets. The severity of these consequences highlights the need for increased awareness, education, and preventative measures. Many online platforms recognize the seriousness of cyberbullying and have implemented policies, reporting tools, and educational programs to create a safer online environment and fight this widespread probl
em.

Hate speech

The spread of hateful speech online is a worrying trend that threatens open and inclusive discussions on digital platforms. This type of expression uses discriminatory language or content to threaten, insult, or incite violence against people based on their race, religion, ethnicity, or other identities. Hate speech not only reinforces negative stereotypes but also creates a hostile online environment. The harm can spill over into the real world, leading to violence and fostering fear and division. To combat this, both online platforms and users need to take action. Platform moderators can develop clear policies against hate speech, create user reporting tools, and use advanced technology to identify and remove hateful content. Additionally, promoting digital literacy and encouraging empathy and respect within online communities are crucial. By working together, we can create a safer and more tolerant online space where diverse viewpoints can be shared without fear of discrimination or violence. Striking a balance between free speech and

protecting individuals and communities from online hatred is a complex challenge, but it's essential in today's digital world.

Targeted Groups

I will deliberate the most prevalent targeted groups as the below:
Gender and Misogyny-This category encompasses any expression of hostility directed at a specific gender or the reduction of individuals based on their gender. It includes posts that are offensive towards a particular gender and encompasses various manifestations of misogyny. Additionally, misogynistic speech, characterized as hate speech specifically targeting women, has emerged as a growing concern in recent years
In the **religious** category, it involves any form of acumen related to religion, such as targeting Islamic sects, promoting atheism, displaying anti-Christian sentiments, or expressing bias against Hinduism and other religions. This may involve upsetting individuals based on their membership in a specific tribe, region, or country.Highlight that religious hate is identified as a motivating factor for crimes in countries experiencing a high prevalence of social crimes.
The **racism** category pertains to any type of racial offense, speech tribalism of a group , regionalism of a person, xenophobi as especially directed at migrant workers and nativism involving hostility towards immigrants and refugees. This category also encompasses prejudice against a particular tribe or region. An example includes causing distress to an individual based on their affiliation with a specific tribe, area, or country, or displaying bias towards a particular tribal identity.

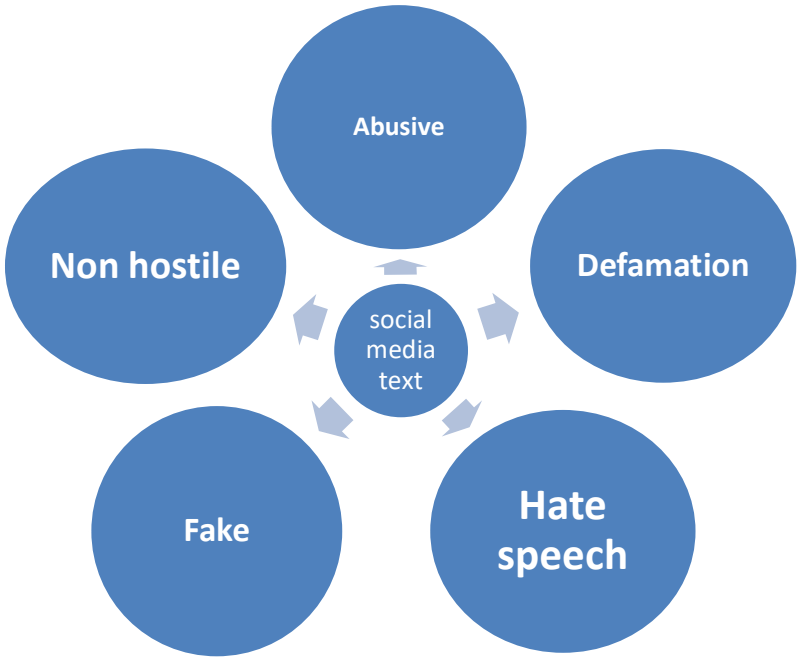

Figure 1

5.8 Type of prosocial behaviour:

Online prosocial behavior refers to positive actions and interactions exhibited by individuals within digital environments, including social media platforms, online communities, and virtual

spaces.. This behavior includes acts of kindness, support, empathy, and cooperation demonstrated through various online activities, such as sharing helpful information, providing emotional support to others, participating in charitable initiatives, and promoting social causes. Online prosocial behavior can take many forms, including offering words of encouragement to peers, volunteering time and resources to assist others in need, or spreading awareness about important social issues. These actions contribute to the cultivation of a supportive and inclusive online community, fostering connections and relationships based on mutual respect and cooperation. Additionally, online prosocial behavior can have a positive ripple effect, inspiring others to engage in similar acts of kindness and contributing to the overall well-being and positivity of digital spaces.

Information Sharing.

Information sharing is a crucial form of prosocial behaviors that enhances the collective knowledge and well-being of communities. When people share information—whether it is data, personal experiences, or advice—they enable others to make better decisions, solve problems, and learn new things. This behaviour promotes a culture of cooperation and trust, strengthening social bonds among community members. In online environments like forums and social media, information sharing can make knowledge more accessible, empower underrepresented groups, and facilitate collective efforts on important issues. By focusing on the distribution of valuable and accurate information, individuals not only meet the immediate needs of others but also contribute to the long-term development and resilience of their community.

Gratitude:

Gratitude is a powerful form of prosocial behaviour that fosters positive relationships and enhances social cohesion. When individuals express gratitude, they acknowledge the kindness and efforts of others, which strengthens bonds and encourages continued generosity. This expression of thankfulness not only benefits the recipient by making them feel valued and appreciated but also positively affects the person expressing gratitude by promoting a sense of well-being and satisfaction. In various settings, whether personal or professional, gratitude can improve interactions, boost morale, and create a more supportive and collaborative environment. By recognizing and celebrating the positive actions of others, gratitude helps build a community where people feel connected and motivated to help one another.

Esteem Enhancement:

Esteem enhancement is a notable form of prosocial behaviour that plays a crucial role in fostering individual confidence and social harmony. By providing praise, recognition, and positive feedback, individuals can significantly boost the self-esteem of others. This encouragement not only helps people feel valued and competent but also motivates them to continue striving for success and contributing positively to their community. Esteem enhancement creates an environment of mutual respect and support, where people are more likely to collaborate and assist each other. In both personal and professional contexts, recognizing and affirming the strengths and achievements of others can lead to increased productivity, better relationships, and a more cohesive social fabric. Through esteem enhancement, individuals contribute to a culture of positivity and empowerment, making communities stronger and more resilient.

Social Support.

Social support is a fundamental form of prosocial behaviour that significantly enhances individual well-being and community resilience. By offering emotional, informational, or practical assistance, individuals can help others navigate challenges and stressors more effectively. This support fosters a sense of belonging and security, encouraging people to reach out and connect with one another. In times of crisis or everyday difficulties, social support can alleviate feelings of isolation and anxiety, promoting mental and emotional health. Additionally, a strong network of support contributes to a cooperative and caring community where individuals feel valued and understood. By actively providing social support, people help build a nurturing environment that benefits everyone, fostering a culture of empathy and solidarity.

Social Cohesion.

Social cohesion represents a vital aspect of prosocial behaviors, essential for creating unified and harmonious communities. It involves actions that promote solidarity, mutual trust, and a sense of belonging among individuals. When people engage in activities that foster social cohesion—such as community events, collaborative projects, or simply offering support to neighbors—they help to build stronger, more resilient social networks. This collective effort not only enhances individual well-being but also strengthens the fabric of society, making it more capable of facing challenges and changes. Social cohesion reduces conflicts and fosters cooperation, creating an environment where people work together for the common good. By prioritizing social cohesion, individuals contribute to a more inclusive and supportive community, ensuring that everyone feels connected and valued.

Fundraising and donating.

Fundraising and donating represent significant forms of prosocial behaviors that have a profound impact on individuals and communities alike. When individuals engage in fundraising efforts or contribute donations, they provide vital support to various causes, organizations, and individuals in need. These actions not only address immediate needs such as financial assistance for medical expenses, disaster relief, or educational programs but also contribute to broader social and environmental initiatives. By pooling resources and rallying support, fundraisers and donors help to create positive change, promote social justice, and address systemic issues. Moreover, fundraising and donating foster a sense of empathy, altruism, and interconnectedness among people, reinforcing the bonds of community and solidarity. Whether it's through organizing charity events, volunteering time, or making financial contributions, individuals who participate in fundraising and donating play a crucial role in building a more compassionate and resilient society.

5.9 Transformer [34]

The NLP transformer represents a groundbreaking model architecture that has transformed NLP tasks. Unlike conventional sequence-to-sequence models, self-attention mechanisms is used in transformer to process input data concurrently, allowing for more effective handling of long-distance relationships. This architecture has notably enhanced the accuracy of various NLP tasks, such as machine translation of sentences , text summarization of a document, and language comprehension. A major advancement of transformers lies in their integration of multi-head attention, permitting the model to work to various parts of the input text concurrently. This concurrent processing capability, combined with efficient training methods like pre-training and fine-tuning using extensive language datasets, has resulted in modern performance across many NLP standards. Exemplified by models such as pretrained BERT and pre trained GPT the transformer architecture continues to propel advancements in both natural language understanding of texts and generation of text respectively.

5.10 BERT

Bidirectional Encoder Representations from Transformers stands as a pioneering model within the realm of NLP, crafted by Google AI. It revolutionized the domain by introducing an innovative technique to pre-training language representations through transformers, thereby enabling bidirectional comprehension of text. Unlike its predecessors, which processed text unidirectionally (such as left-to-right or right-to-left), BERT utilizes masked language modeling to predict omitted words within a sequence, capturing contextual cues from both the directions. This feature equips BERT with a deeper grasp of language semantics and syntax. BERT, pre-trained on huge text datasets, can be fine-tuned for different NLP tasks such as sentiment analysis of texts , named entity recognition of terms , and question answering. It consistently performs exceptionally well across different benchmarks. BERT's success has led to advancements in

transfer learning and has become fundamental in creating various language understanding applications.

BERT's Architecture

BERT, Comprising several layers of self-attention and feedforward neural networks. BERT employs a bidirectional technique to grasp contextual intricacies from both previous and subsequent words within a sentence. The pre-trained versions of BERT vary in scale, with four distinct types based on the size and complexity of the model architecture:

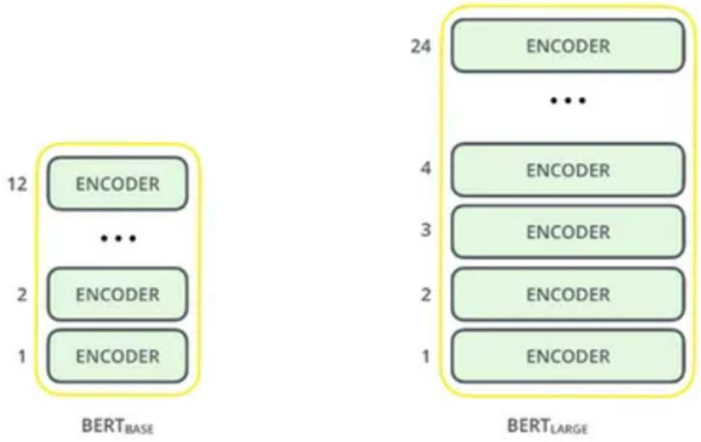

Figure 4

BERT operates through a dual process of pre-training in unsupervised way and fine-tuning in supervised way. This approach involves two primary stages: text preprocessing followed by pre-training tasks.

Text Preprocessing:

In the BERT architecture, the foundational Transformer incorporates an encoder tailored for textual input processing. Given BERT's main objective of establishing a comprehensive language model, only the encoder component is utilized. The input to the BERT involves converting a sequence of tokens into vectors, which undergo processing by the neural network. This process commences with the incorporation of three types of embeddings for each input:

- Token Embeddings: Tokens are augmented with a first token at the beginning of the first sentence, followed by tokens to denote the start and end of each sentence.
- Segment Embeddings: Every token is categorized into Sentence 1 or Sentence 2, aiding the encoder in distinguishing between different sentences within the input.

Positional Embeddings: These embeddings signify the relative position of each token within its sentence, providing essential positional information to the model during processing.

BERT has effectively tackled two NLPtasks:

1. Masked Language Modeling: This task revolves around predicting the next word in a sequence by concealing certain input tokens. Tokens marked as [MASK] are randomly selected for prediction instead of the subsequent token in the sequence.

2. Next Sentence Prediction: This task involves determining if the second sentence in a pair logically follows the first, constituting a binary classification task. This aspect is pivotal for Subsequent uses in applications such as Question Answering of text and Natural Language .

5.11 Performance Evaluation Measures

The field of NLP employs various measures the performance of algorithms is evaluated, precision, recall, and F1-score are few of them. These metrics are defined based on a retrieved documents set and a relevant documents set, such as those generated by a web search engine for a given query. To calculate these measures, words or terms are taken into account, and their values range between 0 and 1.

Precision assesses the ratio of relevant documents retrieved in response to a query. In the context of a web search, it denotes the number of correct retrieve divided by the total number of retrive returned. While a precision score of 1.0 signifies that all retrieved results are relevant, it does not ensure that all relevant documents is retrieved.

$$\text{Precision(p)} = \frac{\text{Relivant documents} \cap \text{Retrived documents}}{\text{Retrived documetns}}$$

Recall assesses the percentage of pertinent dataset successfully retrieved. In text searches, it represents the ratio between accurate results to the entire number of results expected. While a recall score of 1.0 indicates that all relevant documents were recovered, achieving this score by returning all documents in response to the query may not be ideal. Therefore, it is crucial to evaluate both recall and precision jointly when assessing information retrieval algorithms.

$$\text{Recall(R)} = \frac{\text{Relevant Documents} \cap \text{Retrived Documents}}{\text{relevant documents}}$$

The F1-score combines precision and recall into a single measure to assess performance. It is calculated as two times the product of precision and recall, divided by their sum. The F1-score value lies from 0 to 1, where 1 indicates excellent performance and 0 indicates poor performance. A lower F1-score indicates insufficient precision and recall.

$$\text{F1 score} = \frac{2.P.R}{P+R}$$

6. PROJECT TIME LINE

This project has been executed on two phases.

6.1 PHASE 1: An anti-social behaviour detection system has been designed to identify and mitigate harmful behaviours exhibited in various contexts, including online platforms, public spaces, and organizational settings. This system Utilizes state of the art technologies like NLP and machine learning, machine learning, and data set analytics to analyse and interpret patterns of behaviour indicative of aggression, harassment, or misconduct. By detecting and flagging instances of anti-social behaviour, these systems play a crucial role in maintaining safety, promoting positive interactions, and fostering inclusive environments. They are employed across diverse domains, from social media platforms combating cyberbullying to public safety initiatives addressing disruptive behaviours in urban areas

6.2 PHASE 2: Pro-social behaviour has been detected from Hindi text data. Politeness detector and social reward system is developed in this phase. Creating a politeness detector involves developing a system of NLP to identify and gauge the level of politeness in online post. This technology aims to assess language use and expressions, distinguishing between courteous and potentially impolite or offensive content. By integrating with digital platforms, the detector offers a proactive approach to encourage respectful communication. The goal is to enhance online interactions by providing feedback on language choices, fostering a more positive and considerate digital environment.

Objective 1: To develop and implement an effective system for reward in prosocial/antisocial behaviour in Hindi text-based content on various online platforms.

Objective 2: To develop and implement an effective system for identifying and mitigating instances of prosocial behaviour in Hindi text-based content on various online platforms.

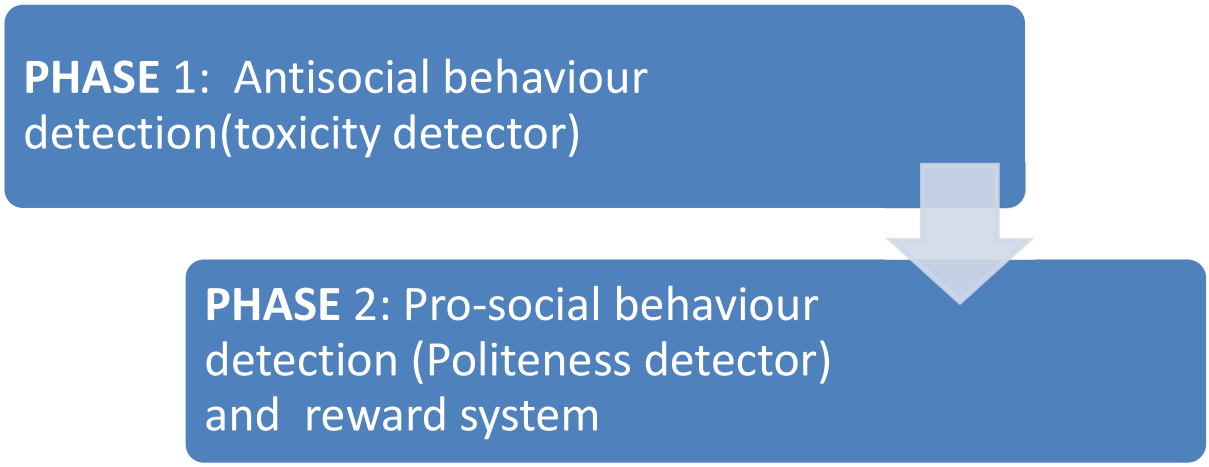

Figure 4

7. PROJECT DESIGN

7.1 Detail design

Multilingual social interaction identifier

The development of a multilingual social interaction identifier constitutes a notable progress in the domain ofonline communication. This innovative tool is designed to discern and categorize social interactions across different languages, facilitating a more inclusive and comprehensive understanding of digital conversations. In an era where individuals from diverse linguistic backgrounds engage in online discourse, the multilingual social interaction identifier plays a crucial role in bridging language gaps and promoting cross-cultural communication. By leveraging advanced language processing technologies, this identifier not only identifies social interactions but also accommodates various languages, enhancing its adaptability to the globalized nature of online platforms. This tool is poised to contribute to the creation of more nuanced and culturally aware algorithms, fostering a digital environment that reflects the richness of linguistic diversity while promoting meaningful and respectful interactions across linguistic boundaries.

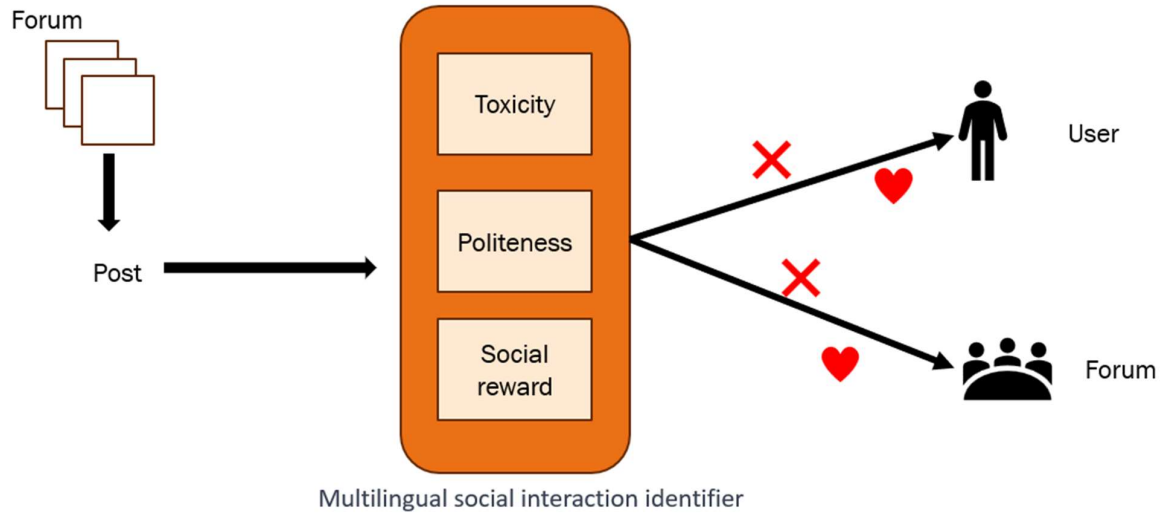

Fig 5

Toxicity detector

The toxicity detector is a vital tool in the digital landscape, playing a pivotal role in safeguarding online spaces from harmful content. Operating on advanced algorithms and machine learning models, this detector identifies and categorizes toxic language which again be divided, such as hate speech, harassment, or offensive comments, within digital interactions. Its implementation is crucial for creating a safer and more comprehensive online environment from text, protecting users from the emotional and psychological impact of toxic content. By swiftly identifying and flagging such behaviour , the toxicity detector enables content moderators and platform administrators to take prompt actions, including content removal or user warnings. The ongoing refinement of toxicity detection algorithms is essential to keep pace with the evolving nature of online interactions and to address emerging challenges. Overall, the toxicity detector is a proactive measure, contributing to the cultivation of a digital space that prioritizes user well-being, fosters respectful discourse, and mitigates the potential harm caused by toxic content.

Politeness detector

The politeness detector characterizes a significant advancement of digital communication tools in the development focusing on fostering respectful and courteous online interactions. Employing sophisticated algorithms and natural language processing techniques, this detector assesses the politeness level of digital content, distinguishing between courteous and potentially offensive language. Its implementation is particularly valuable in mitigating the risk of online conflicts, promoting positive discourse, and creating a more inclusive virtual environment. By identifying impolite language or behaviour s, the politeness detector serves as a proactive measure, allowing platforms to enforce community guidelines and maintain a standard of civility. This tool is crucial in the context of evolving online communication norms, contributing to the enhancement of digital etiquette and encouraging users to engage in conversations with greater consideration and respect for diverse perspectives. As online spaces continue to grow, the politeness detector stands as an essential component in cultivating a culture of respectful dialogue and harmonious interaction within the digital realm.

Social reward/ feedback system

In the realm of online social behaviour, the concept of introducing a reward system holds the potential to profoundly influence the dynamics of digital interactions. By implementing a rewards mechanism, platforms can incentivize positive behaviour s, encouraging users to contribute meaningfully, engage respectfully, and foster a supportive community. Rewards may take various forms, such as badges, recognition, or virtual currency, providing tangible acknowledgments for constructive contributions. This method not only reinforces positive conduct but also creates a sense of achievement and community pride among users. In contrast to punitive measures, which focus on deterring negative behaviour, a well-designed reward system actively cultivates a culture of positivity and collaboration. This strategy aligns with the goal of shaping online spaces into environments where users are not only discouraged from engaging in harmful behaviour but are also motivated to actively contribute to the collective well-being of the digital community. Ultimately, an effective reward system has the potential to transform online social behaviour, fostering a culture that values respect, empathy, and positive engagement.

7.2 Flow diagram:

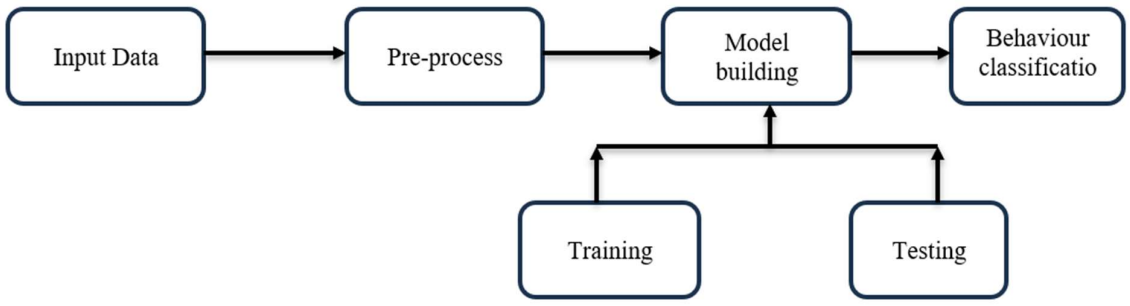

Figure 4

7.3 Algorithm for anti-social and prosocial detector:

Step 1: Import Necessary Libraries

1. Import the libraries such as PyTorch, Hugging Face Transformers, and other utilities.

Step 2: Load and Preprocess the Dataset

1. Load the dataset containing text and labels for prosocial and antisocial behaviour s.
2. Preprocess the dataset: clean the text, handle missing values, and encode the labels.

Step 3: Tokenization and Encoding

1. Initialize the mBERT tokenizer (bert-base-multilingual-cased).
2. Define a function to tokenize and encode the text data using the tokenizer.

Step 4: Prepare Data Loaders

1. The dataset is divided into training, validation, and test sets.
2. Create PyTorch data loaders for batching and shuffling.

Step 5: Initialize the mBERT Model

1. Initialize the mBERT model for sequence classification.
2. The model is moved to the GPU if available.

Step 6: Define the Training Loop

1. Initialize the optimizer = AdamW.
2. Define the loss function = CrossEntropyLoss.
3. Training loop is set to train the model and validate.

Step 7: Evaluation

1. Define the evaluation function to assess on the test data.
2. Performance Metrics like accuracy, precision, recall, and F1-score are calculated.

Step 8: Save and Load the Model

1. The model is saved to disk.
2. Load the model for future predictions.

Step 9: Prediction

1. Define a function to make predictions on new data.

7.4 Pseudo code

1. Import necessary libraries (PyTorch, Hugging Face Transformers, etc.)
2. Load dataset from file
3. Preprocess the dataset:
   - Clean text data
   - Encode labels (0 for antisocial, 1 for prosocial)
4. Initialize mBERT tokenizer with 'bert-base-multilingual-cased'

5. Define function tokenize_and_encode1(text_t, tokenizer, max_length):
   Return tokenizer.encode_plus1(text_t, add_special_tokens=True,
   max_length1=max_length,
   Padding1='max_length', return_attentionn_mask1=True,
   return_tensors='ptt', truncation=True)

6. For each text in dataset:
   Apply tokenize_and_encode to text
   Devide dataset into training, validation, and test sets

7. Define class TextDataset with __init__, __len__, and __getitem__ methods

8. Create train_dataset, val_dataset, and test_dataset using TextDataset

9. Create DataLoader for train_dataset_1, val_dataset_1, and test_dataset with batch_size and shuffle options
   Initialize mBERT model for sequence classification with 'bert-base-multilingual-cased_1' and num_labels=2

10. Model is moved to GPU if available
11. Optimizer is initialized with AdamW(lr=2e-5)

12. loss function is defined as CrossEntropyLoss

13. For each epoch in number of epochs:
    training mode is set
    Initialize total_loss to 0

14. For each batch in train_loader:
    Clear gradients
    Inputs and labels are moved to GPU
    Get model outputs
    Calculate loss
    Perform backpropagation
    Update optimizer
    Add loss to total_loss

15. Calculate avg_train_loss as total_loss / number of batches in train_loader

16. Average training loss is printed for this epoch

17. Set model to evaluation mode
    Initialize total_val_loss to 0
    Initialize empty lists for val_preds and val_labels

18. For each batch in val_loader1:
    Get model outputs
    Calculate loss
    Add loss to total_val_loss
    Append predictions and labels to val_preds and val_labels

19. Calculate avg_val_loss as total_val_loss / number of batches in val_loader
20. val_accuracy is calculated using accuracy_score1(val_labels, val_preds)

21. average validation loss and accuracy is printed for current epoch
    Define function evaluate_model(model, test_loader):
    Evaluation mode is set
    Initialize empty lists1 for test_preds and test_labels

22. For each batch in test_loader1:

Get model outputs
predictions and labels are appended to test_preds and test_labels

23. Define function predict(text, model, tokenizer):
24. Set model to evaluation mode
Tokenize and encode text
Move inputs to GPU
Get model outputs
Return 'prosocial' if prediction is 1 else 'antisocial'

Example usage:
Call predicts with sample text, model, and tokenizer
test_accuracyta is calculated using aaccuracy_score1(ttest_labels, ttest_preds) and Print the prediction result
test_pprecision using precision_scorep(test_labels, test_preds) is calculate
test_recallr using recall_score_1(test_labels, test_preds) is calculate
test_f1 using f1_score_1(test_labels, test_preds) is calculate

test accuracya, precisionp, recallr, and F1 score are printed
Define function evaluate_model(model, test_loader):
model is set to evaluation mode
Initialize empty lists for test_preds and test_labels

For each batch in test_loadertl:
Move labels and inputs to GPU
Get model outputs
add predictions and labels to test_preds and test_labels

Calculate test_accuracy_a using accuracy_scorea(test_labels, test_preds)
Calculate test_precision_p using precision_scoreptest_labels, test_preds)
Calculate test_recall_r using recall_scorer(test_labels, test_preds)
Calculate test_f1 using f1_score(test_labels, test_preds)

Print test accuracy_a, precision_p, recall_r, and F1 score_f
Define function predict(text, model, tokenizer):
Evaluation the model
Tokenize and encode text
Move inputs to GPU
Get model outputs
Return 'prosocial' if prediction is 1 else 'antisocial'

Example usage:
Call predict with sample text, model, and tokenizer
Print the prediction result

7.5 Algorithm for Integrated score:

1. Define Sigmoid Function:
- The sigmoid function is used to normalize the scores.

2. Normalize Scores:

- Use the sigmoid function to standardize the raw scores from the classifiers.

3. Calculate Integrated Score:
- Compute the integrated score using a predefined formula that combines the normalized prosocial and antisocial scores.

Detailed Steps

1. Input:
- Raw score from the prosocial classifier S_p
- Raw score from the antisocial classifier S_a.

2. Define the Sigmoid Function:
sigmoid(x) = $\frac{1}{1+e^{x}}$

3. Normalize the Scores:
- Normalize the prosocial score: S_p
- Normalize the antisocial score: S_a

4. Define the Integrated Score Formula:
- $S_{integrated} = \alpha S_p - \beta S_a$
- Set α and β default to 1.

5. Calculate the Integrated Score:
- Substitute the normalized scores into the formula and compute the result.

7.6 Pseudocode for integrated score:

```
1. FUNCTION sigmoid(x):
2.     RETURN 1 / (1 + EXP(-x))

3. FUNCTION normalize_scores(raw_score):
4.     RETURN sigmoid(raw_score)

5. FUNCTION integrated_score(prosocial_raw_score, antisocial_raw_score, alpha=1.0,
beta=1.0):
6.     prosocial_score = normalize_scores(prosocial_raw_score)
7.     antisocial_score = normalize_scores(antisocial_raw_score)
8.     score = alpha * prosocial_score - beta * antisocial_score
9.     RETURN score

10. prosocial_raw_score = 2.5  // example raw score from prosocial classifier
11. antisocial_raw_score = 1.2  // example raw score from antisocial classifier

12. integrated = integrated_score(prosocial_raw_score, antisocial_raw_score)
13. PRINT "Integrated Score: ", integrated
```

Explanation:

1. Sigmoid Function:
   - The `sigmoid` function's inputs a raw score and normalizes it to a value between 0 and 1.

2. Normalize Scores:
   - The `normalize_scores` function applies the sigmoid function to the raw scores.

3. Integrated Score Calculation:
   - The `integrated_score` function first normalizes the prosocial and antisocial raw scores.
   - It then calculates the integrated score using the formula $S_{integrated} = \alpha S_p - \beta S_a$
4. Example Execution:
   - Example raw scores are provided for prosocial and antisocial behaviour s.
   - The `integrated_score` function is called with these scores, and the integrated score is printed.

This algorithm ensures that the combined score reflects both prosocial and antisocial behaviour s, providing a comprehensive measure of overall behaviour .

8. METHODOLOGIES

8.1 Dataset

Data set for Phase 1:

Political and social issues often lead to significant polarization, with individuals displaying strong emotions tied to their ideologies. This, in turn, results in a notable prevalence of rudeness and hate speech within discussions on platforms like Twitter, particularly those focused on political matters. This corpus primarily comprises tweets centered around political or social topics, uncovering a substantial amount of hate speech. It has been pre-segregated into sub-categories defining the degree of hatefulness in the tweets. This segregation is achieved through a comma-separated values (CSV) file, where each row consists of an index number, the tweet content, and its assigned "hatefulness" category. The categories include **Non-hostile, Fake, Defamation, Offensive, and Hate**. It's significant to note that these categories are not mutually exclusive, leading to various tweets being categorized under multiple labels. For instance, a tweet labeled as "Fake" might also be considered defamatory or offensive. Conversely, tweets marked as "non-hostile" are free of any malicious intent and do not fall into any other categorized labels.

Total 5728 posts are there in this dataset.

non-hostile	0.532472
fake	0.176152
hate	0.083450
offensive	0.070705
defamation	0.053247
hate, offensive	0.028457
defamation, offensive	0.014141
defamation, hate	0.012919
defamation, fake	0.005936
defamation, hate, offensive	0.004888
fake, offensive	0.004888
fake, hate	0.004714
defamation, fake, offensive	0.004190
fake, hate defamation,	0.001571
defamation, offensive fake, hate,	0.001571
hate , fake, , offensive	0.000698

Tab 2

	Unique ID	Post	Labels Set
0	1	मेरे देश के हिन्दु बहुत निराले है। कुछ तो पक्क...	hate,offensive
1	2	सरकार हमेशा से किसानों की कमाई को बढ़ाने के लि...	non-hostile
2	3	सुशांत ने जो बिजनेस डील 9 जून को की थी, वो डील...	non-hostile
3	4	@prabhav218 साले जेएनयू छाप कमिने लोग हिन्दुओं...	defamation,offensive
4	5	#unlock4guidelines - अनलॉक-4 के लिए गाइडलाइन्स...	non-hostile

Figure 6

Dataset for Phase 2:

There are several datasets available that can be used to train classifiers on prosocial behaviour . These datasets typically come from various domains such as social media, forums, and other online platforms where user interactions can be analyzed for prosocial.

For this project I have used a dataset of 11,992 Reddit comments labeled for 5 emotions, including some prosocial emotions like gratitude, admiration, and approval.

8.2 Model Architecture

The BERT architecture is utilized in a project aimed at detecting antisocial and prosocial behaviors in text. By leveraging the pre-trained BERT model, which excels in understanding context and semantics, the project can accurately classify and differentiate between harmful and positive interactions. The model's parameters are fine-tuned specifically for this task, incorporating additional layers tailored to detect nuanced behavioral cues. This setup includes dropout layers for regularization, ReLU activation for non-linearity, and dense layers to process the complex features extracted by BERT. By training on relevant data, the enhanced BERT model can effectively identify and promote prosocial behavior while mitigating antisocial tendencies in online communications.

8.3 Fine-Tune

Fine-tuning BERT for an antisocial and prosocial behavior detection project involves adapting the pre-trained BERT model to classify text according to these behavior categories. By leveraging BERT's deep understanding of language, we can train it on a dataset labeled with examples of antisocial and prosocial behavior. During the fine-tuning process, the model's parameters are updated to optimize its performance on this specific task, enabling it to accurately identify and differentiate between harmful and positive online interactions. This approach enhances the model's ability to analyze and interpret complex language patterns, making it a powerful tool for promoting healthier digital environments.

A training function is used that iterates over batches to performs forward and backward passes of datset. For MLP Specify two concealed layers comprising 30 and 10 neurons respectively, succeeded by a softmax layer. Employ the Rectified Linear Unit (ReLU) is used as the activation function and set the learning rate to lr=2e-5. All other parameters remain at their default settings for each scenario.

8.4 Train the Model

The model training process for the prosocial and antisocial behaviour detector using mBERT involves several crucial steps. Initially, the mBERT model is configured for sequence classification and is moved to the GPU if available, to leverage faster computations. The AdamW optimizer is employed, which is well-suited for handling the huge parameter space typical of transformer models. The difference between predicted and actual labels is computed using cross entropy function. Fine-tuning BERT for an antisocial and prosocial behavior detection project involves adapting the pre-trained BERT model to classify text according to these behavior categories. By leveraging BERT's deep understanding of language, we can train it on a dataset labeled with examples of antisocial and prosocial behavior. During the fine-tuning process, the model's parameters are updated to optimize its performance on this specific task, enabling it to accurately identify and differentiate between harmful and positive online interactions. This approach enhances the model's ability to analyze and interpret complex language patterns, making it a powerful tool for promoting healthier digital environments. The training loop iterates over a predefined number of epochs. During each epoch, the model processes batches of training data, where inputs and labels are moved to the GPU. The model's predictions is computed through forward pass and the loss function quantifies the prediction

error. The using backward pass then adjusts the model parameters to minimize this error. Following every epoch, the model undergoes methode on the validation set to monitor its performance, calculating metrics such as loss and accuracy. This evaluation helps in tracking the model's learning progress and ensures it is to prevent overfitting, the training and validation losses are displayed after each epoch, along with the validation accuracy. The training is limited to three epochs.

8.5 Results

The results of the antisocial and prosocial behavior project revealed significant insights into online interactions. The analysis showed that antisocial behaviors, such as trolling and cyberbullying, are often associated with negative emotional expressions and aggressive language patterns. Conversely, prosocial behaviors, including supportive comments and constructive feedback, were linked to positive emotional expressions and collaborative language. The findings highlight the critical role of fostering a positive online environment to encourage prosocial interactions, which can enhance community well-being and reduce the prevalence of harmful behaviors. These insights can inform the development of interventions and policies aimed at promoting healthier digital communities.

Type of classifier	Precision(P)	Recall(R)	F1 score
Antisocial	0.74	0.85	0.79
prosocial	0.80	0.70	0.74

To combine prosocial and antisocial classifiers into an integrated score, you begin by training each classifier separately to predict prosocial and antisocial behaviour s, respectively. Once trained, normalize the output scores from both classifiers to ensure they are on a similar scale, typically using a sigmoid function to change raw scores to probabilities between 0 and 1. The integrated score can then be calculated using a formula that balances the contributions of both classifiers, such as taking a weighted difference of the normalized scores.

For example, $S_{\text{integrated}} = \alpha \cdot S_p - \beta \cdot S_a$ the integrated score, S_p is the normalized prosocial score, S_a is the normalized antisocial score, and α and β are weights that regulate the importance of each component. These weights can be adjusted based on the desired emphasis on prosocial versus antisocial behaviour. This combined score provides a nuanced evaluation by considering both positive and negative aspects of behaviour, allowing for a more comprehensive classification.

9. CONCLUSION

In conclusion, the project titled "Detection of Online human Behaviour in Hindi Language Posts by Leveraging BERT" represents a significant endeavor in addressing the pressing issue of online antisocial behaviour within the Hindi language context. By harnessing the power of BERT, this project has demonstrated a proactive approach towards identifying and mitigating offensive language, harassment, and other forms of online misconduct.

The utilization of BERT, a cutting-edge language model, has proven to be instrumental in achieving accurate and nuanced detection of antisocial behaviour in the complex linguistic landscape of Hindi. The project's methodology, involving the incorporation of advanced NLP techniques and machine learning algorithms, showcases a robust framework for addressing the challenges posed by online communication.

Through extensive research, data analysis, and model development, the project has paid valuable insights into the patterns and prevalence of online human behaviour in Hindi language posts. The multilingual aspect of BERT has enabled the model to capture the nuances of Hindi expressions, making it an effective tool for monitoring and maintaining a positive online environment.

The findings and methodologies presented in this project not only advance our understanding of online human behaviour but also provide A basis for creating efficient tools and strategies to promote online safety and well-being in multilingual communities. As we continue to maneuver through the ever-changing terrain of digital communication, the lessons learned from this project will undoubtedly inform future endeavors aimed at fostering a more inclusive, respectful, and secure online space for users of diverse linguistic backgrounds.

10. FUTURE SCOPE

The project on "Detection of Online human Behaviour in Hindi Language Posts by Leveraging Multilingual BERT" lays the groundwork for several promising avenues of future research and development. Here are some potential areas for future exploration:

Generate Hindi Prosocial Data: Generating Hindi text for prosocial detection involves creating or sourcing content that exemplifies positive social behaviour . This text can include examples of gratitude, cooperation, and kindness in everyday interactions The focus is on capturing diverse linguistic nuances and cultural contexts relevant to Hindi-speaking communities. This process helps the model learn to identify and classify prosocial behaviour accurately in Hindi text.

Enhanced Multilingual Models: Further refinement and customization of multilingual models, especially tailored to the nuances of Hindi language expressions, could enhance the accuracy and sensitivity of antisocial behaviour detection. Fine-tuning BERT on specific cultural and contextual nuances unique to Hindi could lead to even more effective detection capabilities.

User-Centric Solutions: Future researches may focus on integrating user-centric features, like user history, interactions, and online behaviour patterns, to tailor detection mechanisms to individual users. This personalized approach can lead to more accurate and user-specific identification of antisocial behaviour.

Expanding to Other Indic Languages: Extending the project's scope to encompass a broader range of Indic languages would contribute to a more comprehensive solution. Adapting the model to languages closely related to Hindi and those spoken in the Indian subcontinent could address the diverse linguistic landscape.

11. REFERENCES

1. John Pavlopoulos, Jeffrey Sorensen, Lucas Dixon, Nithum Thain, and Ion Androutsopoulos. 2020. Toxicity Detection: Does Context Really Matter?. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4296–4305, Online. Association for Computational Linguistics.
2. Radford, A.: Improving language understanding by generative pre-training (2018)
3. Bhardwaj, M., Akhtar, M.S., Ekbal, A., Das, A., Chakraborty, T.: Hostility detection dataset in hindi (2020), http://arxiv.org/abs/2011.03588
4. Chowdhury, S.A., Mubarak, H., Abdelali, A., Jung, S.g., Jansen, B.J., Salminen,J.: A multi-platform Arabic news comment dataset for offensive language detection.In: Proceedings of the 12th Language Resources and Evaluation Conferencepp. 6203–6212. European Language Resources Association, Marseille, France (May2020), https://www.aclweb.org/anthology/2020.lrec-1.761
5. Davidson, T., Bhattacharya, D., Weber, I.: Racial bias in hate speech andabusive language detection datasets. In: Proceedings of the Third Workshopon Abusive Language Online. pp. 25–35. Association for Computational Linguistics,Florence, Italy (Aug 2019). https://doi.org/10.18653/v1/W19-3504, https://www.aclweb.org/anthology/W19-3504
6. Vikas Kumar Jha, Hrudya P, Vinu P N, Vishnu Vijayan, Prabaharan P,DHOT-Repository and Classification of Offensive Tweets in the Hindi Language,Procedia Computer Science,Volume 171,2020,Pages 2324-2333,ISSN 1877-0509, https://doi.org/10.1016/j.procs.2020.04.252.
7. Wang, W., Chen, L., Thirunarayan, K., & Sheth, A.P. (2014a). Cursing in English on twitter. *In: S.R. Fussell, W.G. Lutters, M.R. Morris, M. Reddy (eds.) Computer Supported Cooperative Work, CSCW '14, Baltimore, MD, USA, February 15-19, 2014, pp. 415–425. ACM.* https://doi.org/10.1145/2531602.2531734
8. Wang, W., Chen, L., Thirunarayan, K., & Sheth, A.P. (2014b). Cursing in english on twitter. In: Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing, pp. 415–425. ACM
9. F. Shannaq, B. Hammo, H. Faris and P. A. Castillo-Valdivieso, "Offensive Language Detection in Arabic Social Networks Using Evolutionary-Based Classifiers Learned From Fine-Tuned Embeddings," in IEEE Access, vol. 10, pp. 75018-75039, 2022, doi: 10.1109/ACCESS.2022.3190960.
10. Maslej-Kreš ˇ náková, V.;Sarnovský, M.; Jacková, J. Use ofData Augmentation Techniques in Detection of Antisocial Behaviour Using Deep Learning Methods. Future Internet 2022, 14, 260. https://doi.org/10.3390/fi14090260
11. Ojasv Kamal, Adarsh Kumar, Tejas Vaidhya Hostility Detection in Hindi leveraging Pre-Trained Language Models 2021, https://doi.org/10.48550/arXiv.2101.05494
12. Vashistha, N.; Zubiaga, A. Online Multilingual Hate Speech Detection: Experimenting with Hindi and English Social Media. Information 2021, 12, 5. https://dx.doi.org/10.3390/info12010005
13. DevlinJ, ChangMW, LeeK, ToutanovaK. 2019 .BERT:pre-training of deep bidirectional transformers for language understanding . In : Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics :Human Language Technologies, Volume1 (LongandShortPapers). 4171–4186.
14. Davidson, T., Bhattacharya, D., Weber, I.: Racial bias in hate speech and abusive language detection datasets. In: Proceedings of the Third Workshop on Abusive Language

Online. pp. 25–35. Association for Computational Linguistics, Florence, Italy (Aug 2019). https://doi.org/10.18653/v1/W19-3504, https://www.aclweb.org/anthology/W19-3504

15. Bedour Alrashidi, Amani Jamal, Imtiaz Khan and Ali Alkhathlan1 **(2022),** ***PeerJ Comput. Sci.*****, DOI 10.7717/peerj-cs.1142**
16. Hossain, M.Z., Rahman, M.A., Islam, M.S., Kar, S.: BanFakeNews: A dataset fordetecting fake news in Bangla. In: Proceedings of the 12th Language Resources and Evaluation Conference. pp. 2862–2871. European Language Resources Association, Marseille, France (May 2020), https://www.aclweb.org/anthology/2020.lrec-1.349
17. Jain, K., Deshpande, A., Shridhar, K., Laumann, F., Dash, A.: Indic-transformers: An analysis of transformer language models for indian languages (2020)
18. Jha, V.K., P, H., P N, V., Vijayan, V., P, P.: Dhot-repository and classification of offensive tweets in the hindi language. Procedia Computer Science 171, 2324 – 2333 (2020), http://www.sciencedirect.com/science/article/pii/S1877050920312448, third International Conference on Computing and Network Communications (CoCoNet'19)
19. Joshi, R., Goel, P., Joshi, R.: Deep learning for hindi text classification: A comparison. Lecture Notes in Computer Science p. 94–101 (2020), http://dx.doi.org/10.1007/978-3-030-44689-5 9
20. Kakwani, D., Kunchukuttan, A., Golla, S., N.C., G., Bhattacharyya, A., Khapra, M.M., Kumar, P.: IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In: Findings of the Association for Computational Linguistics: EMNLP 2020. pp. 4948–4961. Online (Nov 2020), https://www.aclweb.org/anthology/2020.findings-emnlp.445
21. Kaushal, A., Vaidhya, T.: Winners at w-nut 2020 shared task-3: Leveraging event specific and chunk span information for extracting covid entities from tweets. Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020) (2020). https://doi.org/10.18653/v1/2020.wnut-1.79, http://dx.doi.org/10.18653/v1/2020.wnut-1.79
22. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2017)
23. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized BERT pretraining approach. CoRR abs/1907.11692 (2019), http://arxiv.org/abs/1907.11692
24. Malmasi, S., Zampieri, M.: Challenges in discriminating profanity from hate speech. CoRR abs/1803.05495 (2018), http://arxiv.org/abs/1803.05495
25. Mitrovi´c, J., Handschuh, S.: upinf - offensive language detection in german tweets. In: Proceedings of the GermEval 2018 Workshop 14th Conference on Natural Language Processing (09 2018)
26. Mittos, A., Zannettou, S., Blackburn, J., Cristofaro, E.D.: "and we will fight for our race!" A measurement study of genetic testing conversations on reddit and 4chan. CoRR abs/1901.09735 (2019), http://arxiv.org/abs/1901.09735
27. Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., Chang, Y.: Abusive language detection in online user content. Proceedings of the 25th International Conference on World Wide Web (2016)
28. Ottoni, R., Cunha, E., Magno, G., Bernardina, P., Meira, W., Almeida, V.: Analyzing right-wing youtube channels: Hate, violence and discrimination (2018)
29. Safi Samghabadi, N., Patwa, P., PYKL, S., Mukherjee, P., Das, A., Solorio, T.: Aggression and misogyny detection using BERT: A multi-task approach. In: Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying. pp. 126–131. European Language Resources Association (ELRA), Marseille, France (May 2020), https://www.aclweb.org/anthology/2020.trac-1.20

30. Sreelakshmi, K., Premjith, B., Soman, K.: Detection of hate speech text in hindi-english code-mixed data. Procedia Computer Science 171, 737 – 744 (2020). Https://doi.org/https://doi.org/10.1016/j.procs.2020.04.080,http://www.sciencedirect.com/science/article/pii/S1877050920310498, third International Conference on Computing and Network Communications (Co- CoNet'19)
31. Thorne, J., Vlachos, A., Christodoulopoulos, C., Mittal, A.: Fever: a large-scale dataset for fact extraction and verification (2018), http://arxiv.org/abs/1803.05355 Waseem, Z., Davidson, T.,Warmsley, D.,Weber, I.: Understanding abuse: A typology of abusive language detection subtasks. In: Proceedings of the First Workshop on Abusive Language Online. pp. 78–84. Association for Computational Linguistics, Vancouver, BC, Canada (Aug 2017). https://doi.org/10.18653/v1/W17-3012,https://www.aclweb.org/anthology/W17-3012
32. Wijesiriwardene, T., Inan, H., Kursuncu, U., Gaur, M., Shalin, V.L., Thirunarayan, K., Sheth, A., Arpinar, I.B.: Alone: A dataset for toxic behaviour among adolescents on twitter (2020) Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Brew, J.: Huggingface's transformers: State-of-the-art natural language processing. CoRR abs/1910.03771 (2019), http://arxiv.org/abs/1910.03771
33. Fersini, E., Anzovino, M., & Rosso, P. (2018a). Overview of the task on automatic misogyny identification at ibereval. In: Proceedings of the Third Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2018), co-located with 34th Conference of the Spanish Society for Natural Language Processing (SEPLN 2018). CEUR Workshop Proceedings. CEUR-WS. org, Seville, Spain
34. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin Attention Is All You Need https://doi.org/10.48550/arXiv.1706.03762
35. Architecture Jisu Kim, Curtis McDonald, Paul Meosky, Matt Katsaros, and Tom Tyler(2017) Promoting Online Civility Through Platform , Journal of Online Trust and Safety, September 2022, page 1 of 23 doi:10.54501/jots.v1i4.54
36. Coe, Kevin, Kate Kenski, and Stephen A Rains. 2014. "Online and uncivil? Patterns and determinants of incivility in newspaper website comments." Journal of Communication 64, no. 4 (June): 658–79. https://doi.org/10.1111/jcom.12104. https://doi.org/10.1111/jcom.12104.
37. Jiajun Bao et.al. Conversations Gone Alright: Quantifying and Predicting Prosocial Outcomes in Online Conversations IW3C2 (International World Wide Web Conference Committee), published under Creative Commons CC-BY 4.0 License. ACMISBN978-1-4503-8312-7/21/04. https://doi.org/10.1145/3442381.3450122
38. Rajwal, S. LiHiSTO: a comprehensive list of Hindi stopwords. Multimed Tools Appl (2023). https://doi.org/10.1007/s11042-023-17205-9
39. R. S., V. S., S. T., R. K. and L. Gadhikar, "Vyakranly : Hindi Grammar & Spelling Errors Detection and Correction System," 2023 5th Biennial International Conference on Nascent Technologies in Engineering (ICNTE), Navi Mumbai, India, 2023, pp. 1-6, doi: 10.1109/ICNTE56631.2023.10146610.
40. Harish, B.S., Rangan, R.K. A comprehensive survey on Indian regional language processing. SN Appl. Sci. 2, 1204 (2020). https://doi.org/10.1007/s42452-020-2983-x
41. Pradeepika Verma1, Anshul Verma Accountability of NLP Tools in Text Summarization for Indian Languages ,Journal of Scientific Research, Volume 64, Issue 1, 2020
42. Bhargava, Rupal, Arora, Shivangi and Sharma, Yashvardhan. "Neural Network-Based Architecture for Sentiment Analysis in Indian Languages" Journal of Intelligent Systems, vol. 28, no. 3, 2019, pp. 361-375. https://doi.org/10.1515/jisys-2017-0398

43. Snigdha Paul, Mini Tandon, Nisheeth Joshi and Iti Mathu DESIGN OF A RULE BASED HINDI LEMMATIZER Conference: Third International Conference on Advances in Computing & Information Technology
44. DOI:10.5121/csit.2013.3408
45. Paul, Snigdha et al. "Development of a Hindi Lemmatizer." ArXiv abs/1305.6211 (2013): n. pag.
46. M. Gupta and N. K. Garg, "Text Summarization of Hindi Documents Using Rule Based Approach," 2016 International Conference on Micro-Electronics and Telecommunication Engineering (ICMETE), Ghaziabad, India, 2016, pp. 366-370, doi: 10.1109/ICMETE.2016.104.
47. Harish, B. S. and R. Kasturi Rangan. "A comprehensive survey on Indian regional language processing." SN Applied Sciences 2 (2020): n. pag.
48. Revanuru K, Turlapaty K, Rao S (2017) Neural machine translation of indian languages. In: Proceedings of the 10th annual ACM India compute conference, ACM, pp 11–20
49. M. Mehta et al., "Hindi Text Classification: A Review," 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), Greater Noida, India, 2021, pp. 839-843, doi: 10.1109/ICAC3N53548.2021.9725517.
50. arXiv:2001.10340 [cs.IR] (or arXiv:2001.10340v1 [cs.IR] for this version) https://doi.org/10.48550/arXiv.2001.10340
51. Shalini Puri Satya Prakash Singh An Efficient Hindi Text Classification Model Using SVM
52. DOI: 10.1007/978-981-13-7150-9_24
53. Jain V, Kashyap KL. Ensemble hybrid model for Hindi COVID-19 text classification with metaheuristic optimization algorithm. Multimed Tools Appl. 2023;82(11):16839-16859. doi: 10.1007/s11042-022-13937-2. Epub 2022 Oct 24. PMID: 36313485; PMCID: PMC9589711.
54. Shikha Mundra, , Nikhil Singh and Namita Mittal Fine-tune BERT to Classify Hate Speech in Hindi English CodeMixed Text
55. Dhanashree Kulkarni Sunil S. Rodd Sentiment Analysis in Hindi—A Survey on the State-of-the-art Techniques DOI: 10.1145/3469722
56. Sidhu, S., Khurana, S.S., Kumar, M. et al. Sentiment analysis of Hindi language text: a critical review. Multimed Tools Appl (2023). https://doi.org/10.1007/s11042-023-17537-6
57. Sharma, R., Morwal, S., & Agarwal, B. (2019). Named entity recognition for Hindi language: A survey. Journal of Discrete Mathematical Sciences and Cryptography, 22(4), 569–580. https://doi.org/10.1080/09720529.2019.1637157
58. Richa Sharma, Sudha Morwal, Basant Agarwal, Named entity recognition using neural language model and CRF for Hindi language, Computer Speech & Language, Volume 74, 2022, 101356, ISSN 0885-2308, https://doi.org/10.1016/j.csl.2022.101356.
59. Shelke, R., Vanjale, S. (2023). Review Based on Named Entity Recognition for Hindi Language Using Machine Learning Approach. In: Yadav, S., Haleem, A., Arora, P.K., Kumar, H. (eds) Proceedings of Second International Conference in Mechanical and Energy Technology. Smart Innovation, Systems and Technologies, vol 290. Springer, Singapore. https://doi.org/10.1007/978-981-19-0108-9_35
60. Jain, A., Yadav, D., Arora, A., & Tayal, D. K. (2022). Named-Entity Recognition for Hindi language using Context Pattern-Based Maximum Entropy. Computer Science, 23(1). https://doi.org/10.7494/csci.2022.23.1.3977
61. [Named Entity Recognition for Hindi-English Code-Mixed Social Media Text](https://aclanthology.org/W18-2405) (Singh et al., NEWS 2018)

62. Mundra, Shikha et al. “Fine-tune BERT to Classify Hate Speech in Hindi English Code-Mixed Text.” Fire (2021).
63. Namrata Kumari and Pardeep Singh. 2023. Hindi Text Summarization Using Sequence to Sequence Neural Network. ACM Trans. Asian Low-Resour. Lang. Inf. Process. 22, 10, Article 239 (October 2023), 18 pages. https://doi.org/10.1145/3624013
64. Kumar K, Vimal & Yadav, Divakar & Kumar, Arun. (2015). Graph Based Technique for Hindi Text Summarization. 339. 10.1007/978-81-322-2250-7_29.
65. Bafna, Prafulla & Saini, Jatinderkumar. (2021). Scaled Document Clustering and Word Cloud-Based Summarization on Hindi Corpus. 10.1007/978-981-15-6353-9_36.
66. Namrata Kumari Pardeep Singh Hindi Text Summarization using Sequence to Sequence Neural Network
67. Alrashidi B, Jamal A, Khan I, Alkhathlan A. A review on abusive content automatic detection: approaches, challenges and opportunities. PeerJ Comput Sci. 2022 Nov 9;8:e1142. doi: 10.7717/peerj-cs.1142. PMID: 36426250; PMCID: PMC9680866.

LIST OF PUBLICATION AND THEIR PROOF

IRAJ RESEARCH FORUM

CONFERENCES

CONFERENCE ACCEPTANCE LETTER ISO 9001:2008

I.R.A.J (A Unit of Peoples Empowerment Trust)

INNOVATION PAR EXCELLENCE

CONFERENCE NAME: International Conference on Smart Technology, Artificial Intelligence and Computer Engineering (ICSTAICE-2024)
DATE: *9th May 2024*
VENUE: *New Delhi*
COUNTRY: India

OFFICAL MAIL ID OF CONFERENCE: papers.iraj@gmail.com

Dear Researchers,

Greetings from IRAJ RESEARCH FORUM & Many Congratulations to you!!!!

We are happy to inform you that, your Manuscript **"PERFORMANCE EVALUATION OF LSA ON PILOTS' QUALITATIVE DATA"** has been selected for **ICSTAICE** to be held 9th May 2024 at ***New Delhi, India*** which will be organized by **IRAJ Research Forum** in association with Institute of Research and journals for presentation at the Conference. A Conference Proceeding having **ISBN** *(International Standard Book Number)* and certificates of paper presentation will be given during the event.

UNIVERTIAL PAPER ID (Must use in future Communication)	PAPER TITLE	AUTHOR'S NAME	LAST DATE OF REGISTRATION
IR-AICE-DELH-090524-3184	PERFORMANCE EVALUATION OF LSA ON PILOTS' QUALITATIVE DATA	MISTU MAHAJABIN	Registration Completed

Registration Fees (categories)

AUTHOR	ACADEMICIAN /INDUSTRALIST/ PROFESSORS/	STUDENT			ATTENDEES/LISTENER (With out paper presentation and publication)
		PhD/Post Doc.	M-Tech/ ME/Masters/MBA/MSC/ MBBS/Etc.	B-Tech/BE/ Bachelors/Etc.	
INDIAN	INR 6500	INR 5500	INR 5000	INR 4000	INR 1500
NON INDIAN	USD. 400	USD. 350	USD. 300	USD. 250	USD. 200

Additional value added services fee details	INTERNATIONAL	INDIAN
EXTRA CERTIFICTATE FOR EACH CO-AUTHOR	USD 50	INR 300
CERTIFICATE AND PROCEEDING COPY FOR EACH CO-AUTHOR	USD 100	INR 1000
LUNCH FOR ADDITIONAL GUEST	USD 150	INR 700
EXTRA CERTIFICATE, PROCEEDING AND CONFERENCE LOGO BAG FOR EACH CO-AUTHOR	USD 150	INR 1500

STEPS OF REGISTRATION

STEP-1	STEP-2	STEP-3	STEP-4	STEP-5	STEP-6
Note your *Universal paper ID* **from Acceptance letter**	Select your categories ***(Academician/ Student (M-Tech/PhD)/ Student B-tech)/IISTNER*** form acceptance letter.	Proceed for payment through ***online transfer/NEFT/Cash deposit at Bank*** only to the Bank details mentioned in Mode of Payment(See Below)	Send the Scanned copy of ***Registration form*** (available on conference website) along with ***Bank transaction Details*** to the official **EMAIL-ID** only of the Conference before last date of Registration.	Wait for ***conformation mail*** from Conference coordinator within **one working day** (kindly call to Our Conference Coordinator if any difficulties)	**REGN. COMPLETE** **Wait for Final mail for the Venue and schedule confirmation**

MODE OF PAYMENT

Offline : Bank account details(for NEFT/RTS/Online Banking)	Online: Official Link (click the link)
State bank of India ,Account Name: **Institute of Research and Journals** A/c No. **33547315754** IFSC CODE : **SBIN0010927** SWIFT CODE: **SBININBB270 (For International users)** Bank Address - **SBI, Khandagiri, BBSR, Odisha, India**	http://iraj.in/all_payment.php (Payment by using Debt/Card/Net banking) *(Kindly contact us if any problem during payment processing)*
KINDLY READ CAREFULLY BEFORE REGISTRATION	**DOWNLOAD THE REGISTRATION FORM HERE:**
http://iraj.in/rules.php	http://iraj.in/conf_include/pdf/reg_form_iraj.pdf

Page 1 of 2

Transaction with reference id 526587956 processed successfully.

RRN :	412918338265
Reference ID:	526587956
Debit Account Number:	0991000100503709
Beneficiary Nickname:	IRAJ
Beneficiary Bank Name:	STATE BANK OF INDIA
IFSC Code:	SBIN0010927
Amount(INR):	INR 5,000.00
Transaction Date(dd/MM/yyyy)	08/05/2024
Remark:	payment for conference

NATIONAL CONFERENCE

NCBA - 24

NATIONAL CONFERENCE ON BIGDATA ANALYSIS (NCBA - 24)

2nd June 2024 Kolkata, India

Acceptance Letter

Authors Name: Mistu Mahajabin, Prof. Rajni Jindal

Dear Authors,

We are pleased to inform you that your paper has been accepted by the review committee for Oral / Poster Presentation at the **NATIONAL CONFERENCE ON BIGDATA ANALYSIS (NCBA - 24)**

Article Title: Hindi language processing: A survey

Paper ID: National Conference_8470295

This conference will be held on **2nd June 2024 in Kolkata, India**

Your paper will be published in the conference proceeding and Well reputed journal after registration.

Please register as soon as possible in order to secure your participation:

https://www.nationalconference.in/event/registration.php?id=2482061

You are requested to release the payment and mail us the screen of successful payment release with your name and title of paper to confirm your registration.

Sincerely,

NATIONAL CONFERENCE NC

Dr. Tara Srivastava
National Conference

+91 96770 07228 info@nationalconference.in www.nationalconference.in

Success

Your payment has been made

Dear Mistu Mahajabin,

We are pleased to inform you that your online payment transaction has been successfully processed. The details of your payment are as follows:

Order ID: 664b174ee5513

Reference ID: 113294572132

Amount: 4875.00 INR

Name: Mistu Mahajabin

Email ID: mistu.mahajabin@gmail.com

Phone Number: 918595356545

Thank you for choosing **National Conference**
. We appreciate your prompt payment.

For assistance or inquiries, feel free to reach out to us at **info@nationalconference.in** or **+91 9677007228**.

Best Regards,
National Conference
info@nationalconference.in
+91 9677007228

PLAGIRISM REPORT

PAPER NAME

may25MP2.pdf

WORD COUNT	CHARACTER COUNT
16975 Words	103125 Characters
PAGE COUNT	FILE SIZE
46 Pages	1.6MB
SUBMISSION DATE	REPORT DATE
May 24, 2024 10:34 PM GMT+5:30	May 24, 2024 10:35 PM GMT+5:30

5% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 3% Internet database
- 3% Publications database
- Crossref database
- Crossref Posted Content database
- 4% Submitted Works database

Excluded from Similarity Report

- Bibliographic material
- Quoted material
- Cited material
- Small Matches (Less then 8 words)

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

PLAGIARISM VERIFICATION

Title of the Thesis **Detection of online human behaviour in Hindi language posts by leveraging Multilingual BERT**

Supervisor (s) **Prof. Rajni Jindal**

Department **Computer Science and Engineering**

This is to report that the above thesis was scanned for similarity detection. Process and outcome is given

below:

Software used: **Turnitin** Similarity Index: **5%** , Total Word Count: **16975**

Date:

Candidate's Signature

Signature of Supervisor

BRIEF PROFILE

I am Mistu Mahajabin, pursuing my MTech in Computer Science and Engineering from Delhi Technological University. Currently, I am in the final semester of my degree and I scored 8.4 CGPA in the first three semesters of my MTech.

I completed my BTech in Computer Science and Engineering from West Bengal University of Technology in 2006. After that, I Joined DRDO as scientist B in 2009.Currently I am working at Defence Institute of Psychological Research (DIPR) ,Delhi as Scientist D.

My area of interest is data science, Brain Computer Interface, and Natural Language Processing. I have worked with NLP in many DRDO projects.