الجامعة الإسلامية العالمية ماليزيا
INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA
(Company No. 101067-P)
Garden of Knowledge and Virtue

TAWHIDIC EPISTEMOLOGY LEADING THE WAY
UMMATIC EXCELLENCE LEADING THE WORLD
KHALĪFAH • AMĀNAH • IQRA' • RAḤMATAN LIL-ĀLAMĪN

KICT
KULLIYYAH OF INFORMATION AND COMMUNICATION TECHNOLOGY

Advancement in ICT: Exploring Innovative Solutions (AdICT) Series 3/2025

Editors
Elin Eliana Abdul Rahim
Noor Azura Zakaria
Dini Oktarina Dwi Handayani
Ahmad Fatzilah Misman

KICT Publishing

ADVANCEMENT IN ICT: EXPLORING INNOVATIVE SOLUTIONS (AdICT) Series 3/2025

Editors
Noor Azura Zakaria
Dini Oktarina Dwi Handayani
Elin Eliana Abdul Rahim
Ahmad Fatzilah Misman

KICT Publishing

ADVANCEMENT IN ICT: EXPLORING INNOVATIVE SOLUTIONS
(AdICT) Series 3/2025

First published 2024
Third edition 2025
© Copyright by KICT Publishing

All rights reserved. No part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written consent of Kulliyyah of Information and Communication Technology (KICT), including in any network or other electronic storage or transmission, or broadcast for distance learning

Published by

KICT Publishing
International Islamic University Malaysia
53100 Kuala Lumpur, Selangor, Malaysia

e ISBN 978-629-99388-4-2

9 786299 938842

Cataloguing-in-Publication Data

Perpustakaan Negara Malaysia

A catalogue record for this book is available from the National Library of Malaysia

eISBN 978-629-99388-4-2

Kulliyyah of Information and Communication Technology (KICT)

Preface

It is our great pleasure to present Advancement in ICT: Exploring Innovative Solutions (AdICT), Series 3/2025. This publication brings together diverse and forward-looking ideas in ICT, showcasing innovative approaches that not only address current challenges but also shape future opportunities. The contributions in this volume reflect the dedication, creativity and perseverance of KICT lecturers and students, whose hard work continues to push the boundaries of knowledge and application. Their commitment highlights the spirit of innovation that drives both academic inquiry and practical solutions in the ever-evolving field of ICT.

We would like to extend our sincere appreciation to all authors, reviewers, editors and the organising team for their invaluable efforts in making this series possible. It is our hope that this volume will serve as both an inspiration and a useful reference for scholars, students and professionals. By sharing these innovative perspectives and solutions, we aim to contribute meaningfully to the ongoing discourse in ICT, offering fresh insights and encouraging further exploration in advancing the discipline for the benefit of society.

Editors
Elin Eliana Abdul Rahim
Noor Azura Zakaria
Dini Oktarina Dwi Handayani
Ahmad Fatzilah Misman

TABLE OF CONTENTS

No.	Content	Page No.
1	**School Management System: i-Track** *Aidrina Mohamed Sofiadin, Nur Farah Iwani Jamsari, Nur 'Aqilah Zaidon*	1
2	**Haruka Yum!: Reducing Customers' Precious time with a Dynamic Mobile Application** *Aidrina Mohamed Sofiadin, Khairina Maisarah Husny*	7
3	**It's that Time of the Month!: Development of Digital Intervention for Adolescent Menstrual Health Education** *Hannah Sabrina Saiful Bahri, Hazwani Mohd Mohadis*	13
4	**Cyberpals: A Web-Based Platform for Enhancing Cybersecurity Awareness in Malaysia** *Nur Huda Eshaifol Azam, Nuur Nadheerah Mohammad Lutfi, Andi Fitriah Abdul Kadir*	20
5	**KiddieWheels: Safe, Smart and Simple School Runs** *Nurulhanani Mohd Helme, Nurul Izzah Roslan, Hafizah Mansor*	27
6	**Pomodoro Pro: A Gamified Time Management Mobile Application** *Suhailah Abdullah Zawawi, Nur Adlin Zahid, Nurazlin Zainal Azmi*	34
7	**Smart Personal Assistant** *Nooralya Qasrina Zuraimi, Raihanatuzzahra Azmi, Nurazlin Zainal Azmi*	40
8	**MyBloodBuddy: Blood Donor Mobile Apps** *Nor Hidayati Munadi, Ainin Sofiya Adnan, Noor Azizah Mohamadali*	47
9	**MedTrack: Medication Adherence System** *Adibatunnisa' Ahmad Anuar, Noor Azizah Mohamadali*	53
10	**ComiCraft: Where Arts Meets Community** *Raja Muhammad Hazwan Raja Azman, Omar Haziq Daniel Che Othman, Nor Azizah Mohamadali*	59
11	**WeCareKids: WeCareKids Parenting System** *Joyce Irene Anthony, Mazidah Aiman Awang, Noor Azizah Mohamadali*	65
12	**McFy: A Mobile Application based Medical Leave Verifier** *Muhammad Aqil Ibrahim, Danish Zahiruddin Khairul Anuar, Noor Azizah Mohamadali*	71
13	**CYBERSENSE: Chrome Extension For Real-Time Cyberbullying Detection** *Raini Hassan, Akmal Nazim Mohd Naufal, Imran Hazim Abdullah Salim*	77
14	**Hang Dewal: Learn History Through Role Playing Game** *Wan Mohd Nazim Wan Muhamad Saidin, Faizal Akhtar Azhar, Nor Azura Kamarulzaman*	83
15	**HomeWatch: A Sentiment Analytics Dashboard on Malaysia's Affordable Housing Policies** *Nasya Firzana Mohamad Solleh, Madihah Sheikh Abdul Aziz*	89
16	**A Web-based Solution for Value-Based Software Process Tailoring** *Muhammad Amirul Ashraaf Pakasa, Mohd Izzat Ameir Mat Zain, Noor Azura Zakaria*	94

CYBERSENSE: Chrome Extension For Real-Time Cyberbullying Detection

Raini Hassan
Department of Computer Science
International Islamic University Malaysia
Kuala Lumpur, Malaysia
hrai@iium.edu.my

Akmal Nazim Mohd Naufal
Department of Computer Science
International Islamic University Malaysia
Kuala Lumpur, Malaysia
akmalnazim2002@gmail.com

Imran Hazim Abdullah Salim
Department of Computer Science
International Islamic University Malaysia
Kuala Lumpur, Malaysia
imranhazim02@gmail.com

Abstract*— Cyberbullying, the harmful use of digital platforms to harass or humiliate others, has increased significantly since the COVID-19 pandemic, affecting 34% of individuals and causing serious mental health issues. This project, aligned with Sustainable Development Goal (SDG) No. 3 on good health and well-being, focuses on addressing the gap in real-time detection tools for cyberbullying on social media platforms. Using advanced machine learning techniques, a Bidirectional Encoder Representations from Transformers (BERT) model was developed and achieved a high accuracy of 88% in classifying various types of harmful language. The system demonstrates the strength of BERT in analyzing unstructured textual data and handling large datasets effectively, making it more accurate and reliable than traditional methods. Compared to existing solutions, this project uniquely integrates real-time processing for detecting and managing harmful content on any site, especially social media. Future enhancements include expanding the system to analyze images, memes, and videos, adapting it to other browsers, and developing a user registration feature for personalized experiences. This project not only highlights the potential of AI-driven approaches in addressing cyberbullying but also sets the stage for broader applications, ultimately creating safer online spaces for everyone.

Keywords*—cyberbullying, chrome extension, machine learning, textual data, unstructured data analytics

I. INTRODUCTION

A. *Background*

Cyberbullying is the act of intimidating, threatening, or coercing individuals online by using social media, email, text messaging, blog postings, or other digital or electronic techniques. It often involves derogatory, aggressive, or threatening language [1]. It is more harmful than traditional bullying because it can happen anytime, spread rapidly, and often remains anonymous. In Malaysia, cyberbullying is common among teenagers and continues into adulthood if not addressed early. Traditional methods, such as manual review and keyword filtering, are too slow and allow harmful interactions to persist, causing emotional distress. By using machine learning and unstructured data analytics, this project focuses on analyzing textual data to detect the harmful language in real-time.

B. Problem Statement

Cyberbullying occurs persistently in the digital world, where individuals often express their thoughts recklessly, disregarding the emotional impact on others. A study reveals that 34% of internet users have experienced cyberbullying, with teenagers and young adults being the most affected [1]. According to a report, one in three young people in 30 countries have experienced online bullying, with nearly 71% of children and teenagers reporting that social media platforms like Facebook and Instagram are where it occurs most frequently [2]. The consequences are severe, including anxiety, depression, and, in extreme cases, suicidal tendencies. Another study in 2024 mentions that despite its widespread prevalence, social media platforms struggle to combat this issue effectively, as many lack real-time detection tools powered by artificial intelligence [3]. This gap is especially problematic given that most cyberbullying involves unstructured textual data, making it challenging to analyze and classify harmful language accurately. Addressing this gap is critical to creating safer digital spaces for everyone. This research addresses these gaps by developing an AI-powered browser extension that uses Natural Language Processing (NLP) to detect the harmful language in real-time, aiming to create safer digital spaces for vulnerable users.

C. *Project Objectives*

1) To build a machine learning model that can detect cyberbullying in real-time by analyzing and identifying harmful text accurately.
2) To develop a Chrome extension that alerts users instantly when harmful language is detected and suggests appropriate responses.
3) To protect user privacy by processing data locally and including a feedback system to improve the model's accuracy and performance over time.

D. *Project Scope*

The project's scope includes creating a Chrome extension for real-time cyberbullying detection, as well as an alert system that notifies users when harmful language is discovered, along with explanations and ideas for suitable replies. Users will be able to adjust the detection sensitivity, categories of monitored material, and notification alternatives. The project entails training machine learning models on large datasets of cyberbullying language, leveraging unstructured data analytics to process textual data, and constantly updating the models for greater accuracy. The extension will first be created for Google Chrome and will have a simple, visually pleasing interface for quick navigation. It will have a feedback mechanism that allows users to report false positives and negatives, as well as offer performance evaluation.

II. LITERATURE REVIEW

The search revealed several systems currently available that closely resemble the implementation of this project. A few insights from the existing system will be applied to this project, as well as their advantages and disadvantages.

A. Existing System

TABLE I. ADVANTAGES AND DISADVANTAGES OF EXISTING SYSTEMS

System	Category	Advantages	Disadvantages
Bully	Detection Capabilities	Uses BERT models to detect and manage cyberbullying tweets by blurring or labeling harmful content.	Struggled with nuanced or context-dependent cases of cyberbullying
	User Interaction and Interface	User-friendly interface for reporting wrongly predicted tweets by alert users.	The interface may be challenging for users unfamiliar with technical settings.
	Features	Allows customization of detection sensitivity and notification options	The system offers extensive customization and feedback methods that may be overlooked by users who prefer simplicity.
	Platform	Built as a Chrome extension for Twitter	The system's reach is limited as it was originally built for Chrome and Twitter
ReThink	Detection Capabilities	Encourages users to reconsider publishing potentially dangerous posts.	The system's approach may not detect or handle subtle types of cyberbullying or coded language.
	User Interaction and Interface	User-friendly keyboard extension that warns users before sending potentially dangerous texts, with a simple interface layout.	It may lack complexity in user interaction and customization and the lack of explanation might be frustrating for users.
	Features	Focuses on encouraging people to reconsider their messages before publishing.	The system's focus on encouraging users to reconsider their postings limits its capabilities which may reduce the cyberbullying prevention efforts.
	Platform	The system's keyboard extension functions across various social media platforms	The system relies on human engagement for efficacy and is restricted to sites where users interact by typing
DetoxifyAI	Detection Capabilities	Employs advanced machine learning models to accurately detect a wide range of negative messages.	Struggled to keep up with the dynamic and ever-changing nature of abusive language
	User Interaction and Interface	Integrates seamlessly with other platforms via APIs	The system's API depends on the host platform's implementation which leads to unequal experiences across applications.
	Features	Extensive detection capabilities of offensive language and API integration for customization.	Lack of real-time intervention elements crucial for rapid damage avoidance and it may limit its effectiveness in dealing with multimedia content
	Platform	Chrome extension integrates with various systems via APIs to be incorporated into a wide range of applications and services	The system may be complex and reliant on platform collaboration due to required API interaction that may affect the consistency and breadth of its detection capabilities.

B. Research Paper

In their 2020 study, the authors [4] enhanced cyberbullying detection through a graph structure of tweet embeddings. The study develops an online Dynamic Query Expansion process to address class imbalance and automate data collection, comparing SOSNet with traditional classifiers, and showing superior or comparable results. Advantages include fine-grained classification of cyberbullying based on specific characteristics, enhanced detection accuracy with Graph Convolutional Networks, and real-time feedback to curb bullying. However, the implementation is complex and resource-intensive, scalability issues arise with large datasets, and the focus on specific characteristics may overlook other forms of cyberbullying, potentially leading to bias.

Another study in 2024 [5] evaluated the efficacy of four deep learning models which are CNN, Bi-LSTM, GRU, and LSTM in detecting cyberbullying on Twitter. Using a

balanced dataset of over 47,000 tweets, the study found that the CNN model achieved the highest accuracy of 83.10%, outperforming the Bi-LSTM and GRU models. The research highlights CNN's superior performance in pattern recognition and suggests further exploration to enhance model generalizability and handle informal language. Advantages include improved detection accuracy through various models, detailed analysis beyond accuracy metrics, and insights into online aggression. However, the study notes challenges in detecting subtle cyberbullying, complexity in model implementation, and generalization issues across different social media platforms.

One study in 2023 [6] introduced the Smart Language Checker, a tool employing various Natural Language Processing (NLP) techniques, including Bag of Words, N-grams analysis, Word Embedding, RNNs, and LSTM Networks, to detect offensive language on social media platforms. The tool achieves high accuracy and sensitivity, with precision at 94.2% and sensitivity at 90.8%. Advantages include high accuracy through ensemble methods, deep learning models like BERT, and robust NLP techniques. The tool contributes to research by making its corpus available to researchers. However, disadvantages include contextual limitations, language-specific restrictions, balancing content moderation with privacy, and significant resource requirements for advanced models. Despite these challenges, the Smart Language Checker demonstrates substantial potential for improving online safety and advancing offensive language detection.

C. *Discussion*

These comparisons cover the technologies, programming languages, tools, and frameworks implemented in the development of Cybersense, emphasising efficiency, scalability, and compatibility. Python was chosen as the primary programming language because of its simplicity, extensive machine learning libraries (such as Scikit-learn, PyTorch, and TensorFlow), and strong community support. These libraries allow for the rapid development of a variety of machine learning models, including Naive Bayes, SVM, Random Forest, and BERT, which was finally chosen due to its superior performance in interpreting context and managing complicated text.

Natural Language Processing (NLP) methods such as TF-IDF, tokenization, and GloVe embeddings were employed to efficiently preprocess and represent text input. The TF-IDF identified key phrases, whereas GloVe and BERT embeddings captured semantic meaning, resulting in robust text categorization. Frameworks such as PyTorch were used for deep learning models because of their flexibility and dynamic computation graphs, which facilitated experimentation and model optimization. TensorFlow's production-level deployment capabilities enhanced PyTorch.

The project's extension was designed for the Google Chrome platform, exploiting its large user base and developer-friendly APIs. This solution assured compatibility and convenience of use while allowing for seamless interaction with the browser environment. These technologies were chosen to balance real-time performance, flexibility to big datasets, and future scalability, making them perfect for developing an efficient and user-friendly cyberbullying detection system.

III. METHODOLOGY

A. *Introduction*

In this study, a quantitative research approach was adopted to analyze and classify instances of cyberbullying in tweets. The primary goal was to develop a machine learning model capable of accurately identifying different types of cyberbullying from textual data. This approach is justified due to its ability to handle large datasets and produce quantifiable, reproducible results.

This study employs a quantitative research approach to analyze and classify cyberbullying in tweets, aiming to develop a machine learning model capable of accurately identifying various forms of cyberbullying. The dataset, "cyberbullying_tweets.csv," was sourced from Kaggle that contains about 100,000 textual data comprised from X or Twitter and contains two main variables: "tweet_text," which are real social media text, and "cyberbullying_type," categorized into six classes which are age, ethnicity, gender, religion, other_cyberbullying, and not_cyberbullying. Each category contains 16,000 example data or tweets. For instance, the gender class has tweets with derogatory remarks toward gender identity, while the religion class includes content targeting religious beliefs. A balanced sample was maintained to ensure fair representation across categories. To ensure data quality, preprocessing steps were performed, including handling missing values, removing irrelevant content, normalizing text linguistically through lemmatization and contraction handling, and filtering out noise like special characters, repeated punctuation, and short tweets. Cleaned data was stored in a new variable, "clean_tweet," for streamlined analysis.

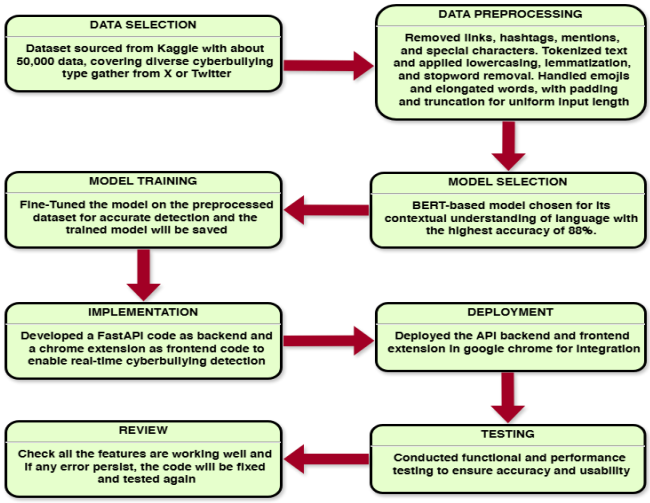

Fig. 1. Methodology

Text representation methods varied by model. Traditional machine learning models such as Naive Bayes, SVM, Random Forest, Logistic Regression, and XGBoost utilized TF-IDF to transform text into numerical vectors. For deep learning models like LSTM, RNN, and BERT, tokenization and embedding techniques were applied. Specifically, GloVe embeddings were used for LSTM and RNN, while the BERT model leveraged the "BertTokenizer" for tokenization and feature extraction. Additionally, the dataset was split into training and validation sets using an 80:20 ratio to evaluate model generalization. Stratified sampling was applied to maintain the original class distribution across both sets. The

model was trained using the AdamW optimizer and a learning rate scheduler, with five epochs to allow the model to converge. A batch size of 16 was used for optimal performance within available GPU memory. Early stopping was implemented based on validation loss to prevent overfitting. The best model was saved for later deployment in the Chrome extension via FastAPI.

The model evaluation involved metrics like accuracy, precision, recall, F1-Score, and confusion matrix analysis, with results visualized using heatmaps to assess performance across different classes. Predictions were tested on new data by implementing preprocessing pipelines, tokenization, and decoding mechanisms to validate model outputs. This comprehensive process, supported by rigorous evaluation, ensures the development of a robust cyberbullying detection system capable of effectively addressing the growing issue of online harassment.

B. *Development Approach*

This study adopted the Agile Software Development Life Cycle (SDLC) approach to develop and refine the cyberbullying detection system. Agile was selected for its iterative nature, flexibility, and focus on continuous improvement, which aligns well with the dynamic requirements of machine learning projects that often necessitate frequent adjustments based on intermediate results.

Fig. 2. Software Development Life Cycle (SDLC)

C. *Requirements Specification*

1) *Functional Requirements*

The system must analyze text in real-time on social media platforms like X or Twitter, ensuring immediate detection of potential cyberbullying. Users should be able to customize notification settings to blur or highlight sensitive content according to their preferences. The system should seamlessly integrate with the Chrome browser, offering an intuitive interface that displays classification results directly on the webpage. Additionally, a dashboard must provide a summary of detected cyberbullying incidents over time, enabling users to interact with the extension's features without interrupting their browsing. All detected incidents, including text, classification, date, and time, must be securely stored in a cloud-based database.

2) *Non-Functional Requirements*

The system must deliver feedback within one to three seconds, ensuring real-time responsiveness, and implement advanced encryption techniques to protect user data during storage and transmission. It should provide a user-friendly interface requiring minimal training, with clear and actionable feedback to help users identify cyberbullying. The system must maintain 99.9% uptime and achieve at least 90% classification accuracy to ensure trust and reliability. Additionally, it must be scalable to support an increasing number of users without compromising performance.

D. *Prototype User Interface*

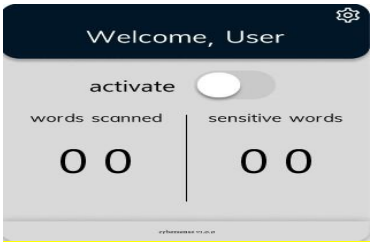

Fig.3 Interface of Cybersense

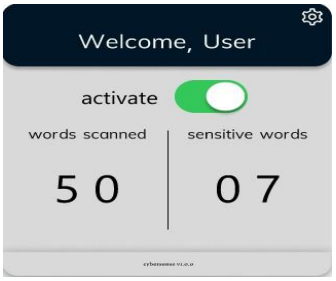

Fig.4 Cybersense when activated

IV. RESULTS

A. *Introduction*

Based on the data collected from various studies and existing systems, the study evaluates machine learning models for detecting cyberbullying, presenting its accuracy or F1-Scores, precision and recall. It achieves the objectives of building a real-time detection model, developing a Chrome extension for alerts and suggestions, and ensuring privacy with local data processing and feedback for improved performance.

TABLE II. TABLE II. RESULTS FOR EVERY TESTED MACHINE LEARNING MODEL

Machine Learning Model	Accuracy	Precision	Recall
Naïve Bayes	0.78	0.76	0.78
Support Vector Machine	0.85	0.86	0.85
Random Forest Classifier	0.86	0.86	0.86
Recurrent Neural Network	0.85	0.85	0.85
Logistic Regression	0.84	0.84	0.84
Long Short-Term Memory (LSTM)	0.84	0.84	0.84
XGBoost	0.85	0.86	0.85

Machine Learning Model	Accuracy	Precision	Recall
Bidirectional Encoder Representations from Transformer (BERT)	0.88	0.88	0.88

The findings show that BERT outperformed other models with the highest accuracy (0.88) and F1-score (0.87), demonstrating its strength in handling unstructured textual data. Random Forest Classifier and XGBoost also performed well, with scores of 0.86 each. Traditional models like Naïve Bayes performed comparatively lower, emphasizing the importance of advanced algorithms like transformers in achieving reliable results. These findings address the research objectives by confirming BERT's effectiveness for real-time cyberbullying detection.

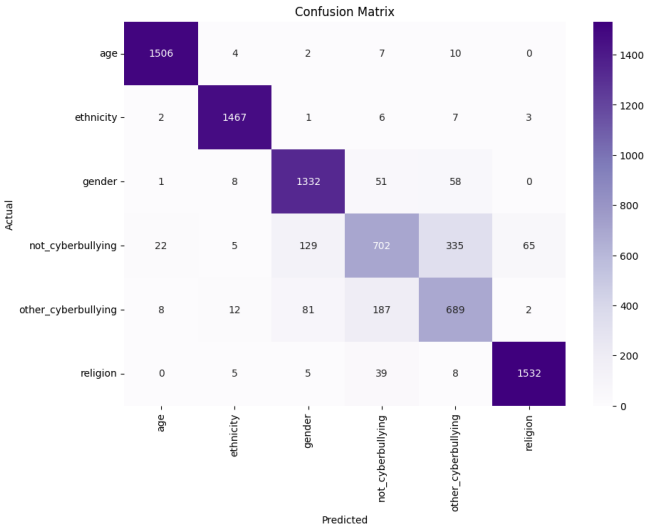

Fig.5 Confusion Matrix for BERT

The confusion matrix highlights the strength of the model in detecting explicit forms of cyberbullying, while also revealing areas where false positives and false negatives occur most frequently. The figure shows that the BERT model correctly predicted most categories, especially for "religion," "age," and "ethnicity," which had high accuracy. However, there were noticeable misclassifications between "not_cyberbullying" and "other_cyberbullying," where the model sometimes confused general content with subtle harmful language. This suggests the model works well for clear cases but may need improvements to better detect borderline or indirect cyberbullying.

The results align with existing literature that highlights BERT's superior performance in natural language processing tasks. Several existing systems and studies provide valuable insights into cyberbullying detection. Some systems utilize machine learning and user-friendly interfaces but face challenges such as limited platforms, nuanced language detection, and real-time intervention. Research papers highlight advancements like graph-based tweet embeddings, deep learning models, and ensemble Natural Language Processing (NLP) techniques for detecting offensive language. While these approaches improve accuracy and offer innovative methods, they also face limitations like resource intensity, scalability issues, and challenges with informal or subtle language. This project confirms BERT's ability to outperform them due to its bidirectional context understanding. These findings expand on existing knowledge by demonstrating BERT's real-world applicability in cyberbullying detection.

B. *System Development Process*

For the development of Cybersense, the system was implemented using a combination of modern tools and frameworks to ensure efficient development and functionality with an Agile methodology adopted to ensure flexibility and iterative progress in creating a real-time cyberbullying detection Chrome extension. The system will be using extensive preprocessing, including tokenization and lemmatization, and preparing the text data for classification. This project implemented the best machine learning model with high accuracy at least 80% to be able to classify every word into its classes correctly. Every model was trained and validated through various metrics including accuracy and confusion matrix visualization to find the best model for this system. The iterative development and testing cycles ensured robust performance and adaptability to user needs, leading to a comprehensive solution for detecting and managing cyberbullying in real-time.

The frontend was developed as a browser extension, primarily using HTML, CSS, and JavaScript. The user interface (UI) was designed to be simple and intuitive, ensuring that users could easily interact with the extension. Key features included real-time text input analysis, with a clear display of results showing whether the input text was identified as cyberbullying. The user experience (UX) was optimized to minimize disruptions, providing seamless interaction while users browse the web. The extension was designed to be unobtrusive yet responsive, offering immediate feedback when harmful content is detected.

The backend logic was implemented using Python with FastAPI, which allowed for the development of a lightweight API to handle requests from the frontend. The main business logic involved processing text input, running it through a pre-trained BERT model for sentiment and emotion analysis, and returning the results. The backend also managed the integration of the machine learning model, ensuring it could handle text analysis in real-time. The API handled all communication between the frontend and the backend, ensuring secure and efficient data processing.

The integration of the frontend, backend, and machine learning model was done by linking the browser extension with FastAPI. When a user types or views any text, the extension sends the data to the API for processing. The backend uses the BERT model to analyze the text and returns the result to the frontend, which then displays whether the input was considered harmful. All components were tested together to ensure seamless communication and ensure that the system functions as a cohesive and fully operational tool for real-time cyberbullying detection.

For real-time interaction, the Chrome extension uses the "chrome.runtime" and "chrome.scripting" APIs to extract visible text from webpages every 3 seconds. This text is split into words and sent to a locally hosted FastAPI server that runs the BERT model. The model processes each word and returns its predicted class. The system dynamically updates the scanned words and classifies them to their classes which can

be viewed in the console log by the user. This setup ensures fast, continuous detection without affecting the browser's performance.

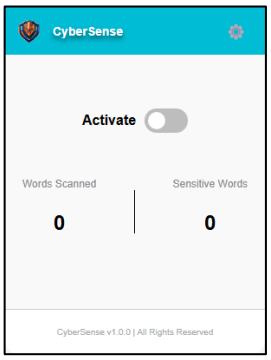

Fig.6 Interface of Cybersense Main Page

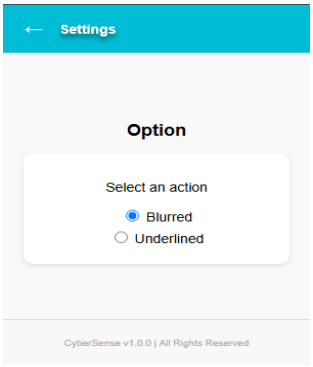

Fig.7 Settings Page for Cybersense

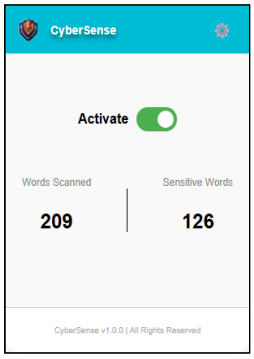

Fig.8 Cybersense when activated

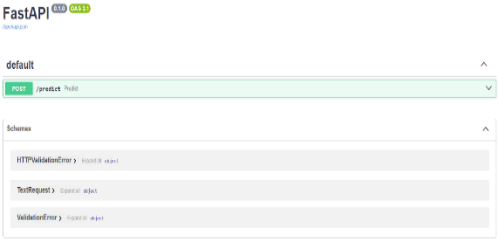

Fig.9 FastAPI System for Backend of Cybersense

Fig.10 Overview of Cybersense

Fig.11 Example of Words Scanned with Class of Cyberbullying

V. CONCLUSION

The system development process for CyberSense followed a structured lifecycle that included design, implementation, and testing phases. The design phase focused on creating a user-friendly interface for the browser extension and outlining the backend architecture to support real-time cyberbullying detection. In the implementation phase, the frontend was developed using HTML, CSS, and JavaScript, while the backend was built using Python and FastAPI, with a BERT model integrated for text analysis. Key milestones included successfully integrating the machine learning model with the Flask API, enabling real-time detection of harmful content in user text. During testing, the system was refined to ensure minimal latency and accurate predictions, overcoming challenges like optimizing performance and ensuring seamless integration between the frontend and backend. A significant achievement was achieving reliable and responsive functionality, with the system accurately identifying cyberbullying content during real-time use, meeting performance targets.

A. *Limitation*

Despite its strengths, the system has some limitations that require attention for future improvement. Although the system is effective in detecting harmful text, it is limited to analyzing textual content and does not support multimedia analysis such as images, memes, or videos. This restricts its capability to detect harmful content in formats commonly shared online. Additionally, the system is designed exclusively for the Google Chrome browser and has not been adapted for other browsers like Firefox, Safari, or Edge. The lack of integration with social media platforms further narrows its reach and usability in broader online environments. The Natural Language Processing (NLP) model used in the system struggles with understanding complex contexts like sarcasm, slang, and multilingual text, which reduces its accuracy in nuanced cyberbullying scenarios.

Furthermore, the system does not provide personalized features such as user registration, saved analysis history, or custom settings, limiting its ability to offer tailored user experiences. The absence of a visual indicator, such as blurring or highlighting harmful words on webpages, also affects the user's ability to engage with flagged content effectively. Lastly, the system's performance is constrained by the dataset used for training, which may lack diversity in terms of demographics, regions, and communication styles. This can lead to biases or reduced accuracy when applied to different user groups. Additionally, the system may face challenges on websites with complex structures or dynamic content, impacting its real-time detection capabilities. Addressing these limitations in future iterations will enhance the system's functionality, accessibility, and overall effectiveness.

B. Recommendations for Future Work

Future improvements to Cybersense could include adding functionality to blur or highlight harmful words on webpages. This feature would allow users to control how harmful content is presented, either by making it less visible to reduce emotional impact or more visible to raise awareness. Implementing this functionality would involve updating the browser extension to dynamically modify webpage content using JavaScript-based text processing libraries. Another key enhancement involves analyzing images, memes, and videos to detect harmful content. By incorporating advanced techniques such as computer vision and deep learning frameworks, the system can identify offensive visual elements like harmful text embedded in images or inappropriate gestures in videos. This would require integrating models like Convolutional Neural Networks (CNNs) or pre-trained models such as YOLO for object detection. The system could also be adapted to work on browsers like Mozilla Firefox, Opera, and Microsoft Edge, as well as on social media platforms through API integrations. This expansion would involve redesigning the extension's compatibility layer and using platform-specific APIs.

Additionally, introducing user accounts for personalized options is another future improvement. Users could save analysis histories, adjust custom settings, and receive tailored feedback by integrating a secure database system like Firebase or MongoDB. The system's Natural Language Processing (NLP) model could also be enhanced to understand better complex language elements like sarcasm, slang, and multiple languages. This would involve training the model with more diverse and context-rich datasets, potentially using transformers like GPT or fine-tuning multilingual models like mBERT. Lastly, expanding the dataset diversity is crucial. Future efforts should focus on collecting data from multiple social media platforms and online communities to ensure the tool is effective across different demographics and regions. Collaboration with organizations and researchers to access anonymized data could accelerate this process. This enhancement could be part of a long-term improvement plan to allow the system to have the best cyberbullying prevention across the digital world.

REFERENCES

[1] N. Latto, "Cyberbullying: What you need to know," Cyberbullying: What You Need to Know, Mar. 13, 2024. https://www.avast.com/c-cyberbullying

[2] UNICEF poll: More than a third of young people in 30 countries report being a victim of online bullying. (2019, September 4). UNICEF. https://www.unicef.org/turkiye/en/press-releases/unicef-poll-more-third-young-people-30-countries-report-being-victim-online-bullying

[3] Philipo, A. G., Sarwatt, D. S., Ding, J., Daneshmand, M., & Ning, H. (2024). Cyberbullying Detection: Exploring datasets, technologies, and approaches on social media platforms. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2407.12154

[4] J. Wang, K. Fu, and C.-T. Lu, "SOSNET: A Graph Convolutional Network Approach to Fine-Grained Cyberbullying Detection," 2021 IEEE International Conference on Big Data (Big Data), pp. 1699–1708, Dec. 2020, doi: 10.1109/bigdata50022.2020.9378065.

[5] V. A. Joseph, B. R. Prathap, and K. P. Kumar, "Detecting Cyberbullying in Twitter: A Multi-Model Approach," Detecting Cyberbullying in Twitter: A Multi-Model Approach, pp. 1–6, Mar. 2024, doi: 10.1109/icdecs59733.2023.10502699.

[6] S. Kavitha, J. V. Anchitaalagammai, S. Murali, R. Deepalakshmi, L. R. Himal, and M. S. Suryakanth, "Smart Language Checker: A Machine Learning Solution for Offensive Language detection in Social Media," 2023 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI), pp. 1–6, Dec. 2023, doi: 10.1109/icdsaai59313.2023.10452454.

e ISBN 978-629-99388-4-2

KICT
KULLIYYAH OF INFORMATION AND COMMUNICATION TECHNOLOGY

KICT Publishing