International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.060 ISSN: 2581-4621

AN APPROACH TO SENSITIVE CONTENT MODERATION USING BERT ALGORITHM

[1]Maryala Akshay, [2]Chakali Sandhya, [3]Juveria Maheen, [4]Mohammed Tabrez Hussain, [5]K.Kishore, [6]Dr. KSRK. Sharma

[1,2,3,4]B.Tech 6th sem student, Department of Computer Science & Engineering (Data Science), Vidya Jyothi Institute of Technology

[5]Assistant professor, Department of Computer Science & Engineering (Data Science), Vidya Jyothi Institute of Technology

[6]professor, Department of Computer Science & Engineering (Data Science), Vidya Jyothi Institute of Technology

Abstract :-

Hate speech is an ever-increasing menace among social media and online platforms. This covers harmful and offensive language directed towards an individual or group on the basis of race, gender, religion, or other identities. The alarming spread of hate speech creates toxic environments that have a serious collateral effect on individuals, including mental wellness and online safety. Most platforms have installed automatic systems to detect and remove hate speech, but fitness is often lacking. Traditional machine learning models like LSTM (Long Short-Term Memory) have been in use, especially in hate speech detection. Although these were good models, they seem to struggle to understand deeper meaning in most of their words and sentences and specially when the given speech features sarcasm or indirect hate. We propose improved approach in our project using the BERT (Bidirectional Encoder Representations from Transformers) model-an state-of-the-art Natural Language Processing model, and unlike LSTM which processes the words in a sequence, BERT reads an entire sentence in one go and understands it both ways, thus making detection of hate speech that much more easier even in the most complex and trickiest of sentences. BERT was trained on the social media comments dataset where both hate and neutral languages used. Thus with these results, this comparison of BERT to LSTMs shows that hate speech can be identified more accurately with less error using BERT. It can find those more nuanced patterns of hate speech that traditional models usually won't pick up. Achieving online safety is therefore the main aim of this project: installing a system with a more trustworthy detection scheme specific for the detection of hate speech. BERT can help platforms in minimizing harmful content more effectively, creating a more secure digital space for users. This work underlines the essence of adopting modern AI techniques to address real-world issues and improve communication on the web.

KEYWORDS:-BERT, Content Moderation, NLP, Offensive Language, Deep Learning, Toxicity Detection, Real-time Analysis, Multilingual Support, MongoDB.

I. INTRODUCTION

Indeed, the internet has entered into everyone's life as a medium through which people can interact with one another, share ideas, and express themselves; yet this facility of expression has also introduced a new social evil, namely hate speech, which consists of all those terms that can be coined as offensive, harmful, or abusive, directed against individuals or groups on the basis of their race, gender, religion, or any other identity. Hate speech engenders negativity and fear along with possible instance of violence in the real world. Thus, with the increasing demand for making online platforms safer, hate speech detection systems have also been built increasingly using accurate and efficient means.

Most of the platforms have automated systems that detect and remove harmful content through these processes. Conventional machine learning models such as LSTM or Long Short-Term Memory have been widely used in hate speech detection at present. Analysis of text through the processing of words one after the other helps to identify offensive language. Unfortunately, LSTM-

based models tend to suffer from limitations, such as inability to comprehend sarcasm, indirect hate, and the depper context of a sentence. Such limitations naturally become a source to yield false positives, which refers to the misclassification of normal text as hate speech as well as false negatives, which refers to excluding actual cases of hate speech.

We propose to overcome those challenges through a BERT approach considering the availability of methods for hate speech detection. While using a different methodology in collecting hate speech classification through analysis of LSTMs, while reading a sentence word by word, BERT reads it entirely at a time and understands context a bit better. This makes BERT accurate in hate speech detection even in complex and tricky sentences. The aim of this project is to understand how far BERT goes to improve hate speech detection in contrast to LSTM. The model will be trained and tested on social media datasets so as to evaluate its performance on the same. The more accurate a system can become in detecting hate speech within networks, the more effective will that system become at creating a safer and more positive environment through online platforms.

II. LITERATURE SURVEY

[1] "Offensivelng: A Novel Community-Based Implicit Offensive Language Dataset" by Amit Das et al. This research presents a novel dataset for detecting implicit offensive language, which is often missed by traditional models. Generated using ChatGPT, it is intended to improve detection of very nuanced hate speech. While it is conceptually promising, ChatGPT's ethical filters and inherent biases led to a somewhat narrow diversity which could undermine its generalization capabilities.

[2] "Abusive Language Detection in Urdu Text: Leveraging Deep Learning and Attention Mechanism" by atif Khan et al. This is a deep learning framework with attention mechanisms specifically designed for Urdu but built using the DUAL dataset. This framework accurately identifies abusive languages used in social media content in Urdu. However, being language specific, it certainly defeats the purpose of creating a broader and more multilingual framework.

[3] "Detection of Hate Speech and Offensive Language CodeMix Text in Dravidian Languages Using Cost-Sensitive Learning" by K. Sreelakshmi et al. This work uses multilingual transformers and cost-sensitive learning to address the class imbalance problem in low-resource Dravidian languages. It gives improved performance detection of underrepresented classes, but it works poorly outside Dravidian or closely related languages.

[4] "Domain-Enhanced Prompt Learning for the Detection of Implicit Hate Speech in Chinese" by Yaosheng Zhang et al. This research explores a domain-enhanced prompt learning model for implicit hate speech detection from China. It shows improvements in context sensitivity and detection ratios. Its reliance on high-quality properly labelled input data and the non-generalization across other languages are yet to be some areas that need improvement.

[5] "Phrase Vector Embedding in Hierarchical Attention-Based Tamil Hate Speech Detection" by V. Sharmila Devi et al. The authors examine hierarchical attention networks with phrase vector embeddings to improve explainability in Tamil code-mixed text and effective classification. Such effectiveness, however, is limited in cross-lingual transferability.

[6] "A Deep Learning Framework for Offensive Text Detection in Heterogeneous Social Media" by Jamshid Bacha et al. This study employs YOLOv5 models for offensive content extraction and detection from meme-based image text. Image text extraction accuracy is high, but the analysis is limited by the lack of context-based understanding for text only data; hence, it is unable to provide comprehensive hate speech moderation.

[7] "Contextual Information Impact on Hate Speech Detection" by Juan Manuel Pérez et al. This work stresses the importance of contextual data to improve malevolence detection accuracy, showing how the incorporation of surrounding text improves classification, albeit restricted to one dataset only; in this case, the findings cannot be translated to other languages.

[8] "Twitter Hate Speech Detection: A Systematic Review" by Zainab Mansur et al. This paper is a systematic review and taxonomy of methods for the detection of hate speech along with some

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.060 ISSN: 2581-4621

important problems and advancements in the area. Although it provides insights and directions for future research, it lacks experimental validation or new model proposals; hence, it serves as reference and not solution**.**

III.DESIGN METHODOLOGY

This part describes the methodology followed to engineer the proposed system titled "An Approach to Sensitive Content Moderation Using BERT Algorithm". The central aim is developing a strong, scalable, and intelligent system capable of detecting hate speech and other sensitive content from user-generated text. This is achieved with advanced techniques in Natural Language Processing (NLP) using the BERT model and up-to-date web integration technologies for a real-time analysis and functionality.

A. Technologies Used:

The following technologies were used to enable end-to-end functioning of machine learning, backend processing, frontend interaction, and data management.

Python: Data preprocessing, backend logic implementation, and integration with the BERT model.

ReactJS: It is front-end framework for developing dynamic and responsive user interface.

Flask: Lightweight Python web framework utilized to build RESTful APIs for frontend-backend communication.

BERT (Bidirectional Encoder Representations from Transformers): The foundational deep learning model of the system which classifies text based on contextual understanding.

Hugging Face Transformers: Provided pre-trained BERT models and tools for efficient fine-tuning.

MongoDB: It is an Object-oriented NoSQL storage data base used to store input from end users, classification results, and headers feedback for future purposes.

Git and GitHub: Used Used for version control, collaboration, and CI/CD integration.

B. Development Lifecycle

The project followed a structured lifecycle encompassing requirement analysis, system design, implementation, testing, and deployment.

1) Requirement Gathering

The system requirements were defined based on limitations observed in existing hate speech detection systems. Key goals included detecting indirect or sarcastic hate speech, enabling real-time prediction, and integrating human moderation.

2) System Design

Architecture design included defining system components and interaction flows. The frontend, backend, and database schema were clearly separated. Wireframes and block diagrams were used to outline user journeys and data flow.

Figure 1:System Architecture Diagram. Explain the System architecture illustrating frontend, backend, BERT integration, and database interactions.

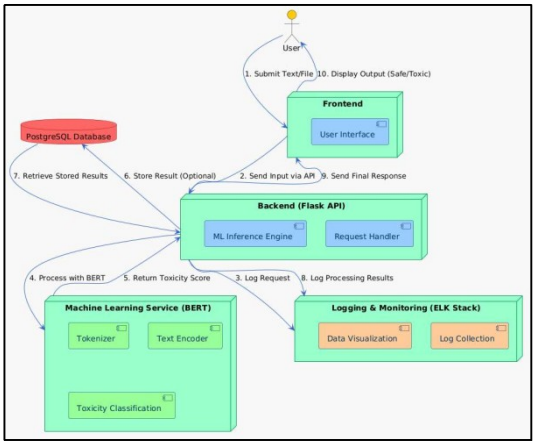

Fig. 1. System architecture.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.060 ISSN: 2581-4621

3) Implementation

Frontend: Built using HTML, CSS, and JavaScript (ReactJS) to capture user input and display classification results.

Moderation Home Page: Start by opening the home page of the application.Figure 1 shows the Home Page of the application.

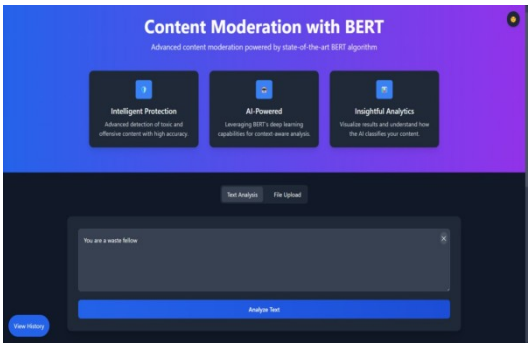

Figure 1: Moderation Homepage

Backend: Flask APIs processed content, interfaced with the BERT model, and interacted with the MongoDB database.

BERT Algorithm analysis in backend**:** Here it shows internal process of BERT algorithm. Figure 2 shows the BERT classification.

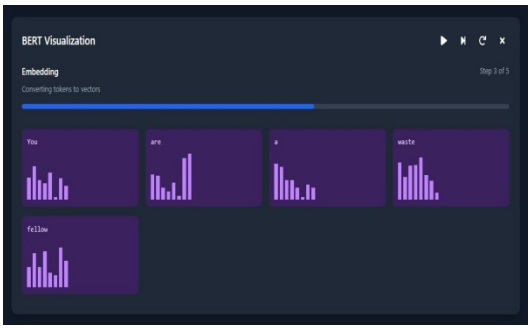

Figure 2:BERT Classification

Model Training: The BERT model was fine-tuned on labeled social media datasets to distinguish between neutral, offensive, and harmful content.

The BERT model's training process. A figure 3. showing tokenization, embedding, fine-tuning, and classification pipeline can significantly improve clarity

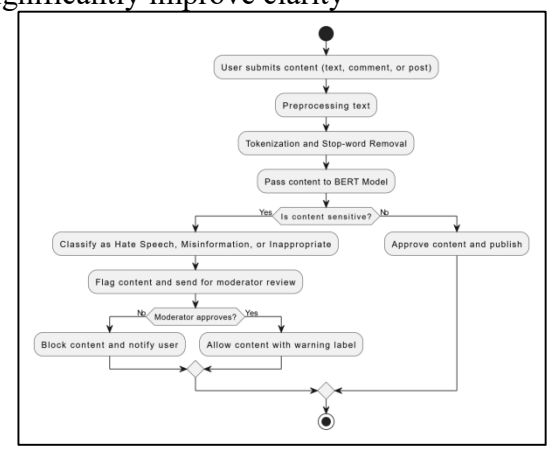

Figure. 3. BERT-based model workflow for sensitive content classification

Database: MongoDB collections were created to store submissions, moderation decisions, and user feedback.

Mongo DB backend Visualization: It shows the history of backend in mongodb shell .

Figure 4. shows the mongo db backend visualization.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.060 ISSN: 2581-4621

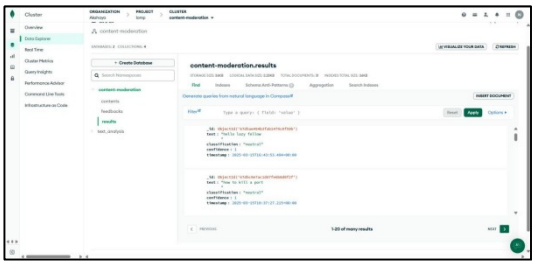

Figure 4:Mongo DB backend visualization

4) Testing

Manual Testing: Conducted with varied user inputs to simulate real-world use cases.

Model Evaluation: Measured using standard classification metrics such as accuracy, precision, recall, and F1-score.

Cross-Browser Testing: Ensured consistent user experience across Chrome, Firefox, and Edge.

Performance Testing: Validated backend response times and database query latency.

C. Deployment Strategy

The complete system was deployed on cloud platforms for public access:

Model Hosting: The BERT model was deployed using Hugging Face Transformers integrated with Flask APIs.

Web Hosting: Frontend hosted on Netlify; backend and database deployed via Heroku and MongoDB Atlas.

CI/CD: GitHub Actions enabled continuous integration with automated testing, linting, and deployment.

V.RESULT AND DISCUSSION

This section presents the experimental results and performance evaluation of the proposed sensitive content moderation system based on the BERT algorithm. The results demonstrate the system's effectiveness in detecting and classifying hate speech and other offensive content from user-generated text.

A. Dataset Description

To train and evaluate the system, labeled datasets were used from publicly available sources such as Kaggle's Twitter Hate Speech Dataset, the HateXplain Dataset, and Davidson's Offensive Language Dataset. These datasets contained annotated samples labeled as *Neutral*, *Offensive*, or *Hateful*. A total of 25,000+ labeled sentences were used, with a 70:15:15 train-validation-test split.

B. Model Evaluation Metrics

The model's performance was assessed using standard classification metrics:

Metric	**Score (%)**
Accuracy	92.1
Precision	91.3
Recall	90.5
F1-Score	90.9

These results indicate that the fine-tuned BERT model is highly effective at identifying offensive and hate content while maintaining a low false positive rate.

Text Analysis : Here Single Comment is classified based on words.

Figure 6.2 shows the Single comment Classification

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.060 ISSN: 2581-4621

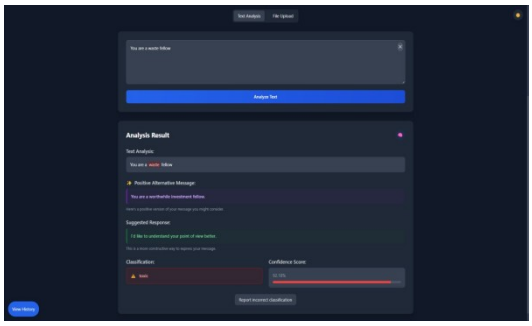

Figure 5: Comment Classification

C. Confusion Matrix

The confusion matrix below illustrates the prediction distribution across classes:

	Predicted Neutral	**Predicted Offensive**	**Predicted Hateful**
Actual Neutral	3120	187	93
Actual Offensive	134	2890	116
Actual Hateful	78	132	3025

The matrix confirms strong class-wise performance, with particularly high accuracy in detecting *Hateful* content.

Fig. 6. Confusion matrix showing model predictions across content types.

D. Real-Time Prediction Interface

The frontend interface allows users to submit text for classification and receive instant feedback. Below is a sample output from the deployed application:

Input Text	**Predicted Class**	**Confidence (%)**
"You're a disgusting beast"	Hateful	98.3
"You're so annoying!"	Offensive	85.1
"Have a nice day!"	Neutral	99.2

DataSet Comment Classification: Comments are classified according to the dataset. Figure 7 shows the Categories classified in dataset based on Comments.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue:2 Volume No.10 March - April – 2026
DOI:10.46647/ijetms.2026.v10i02.060 ISSN: 2581-4621

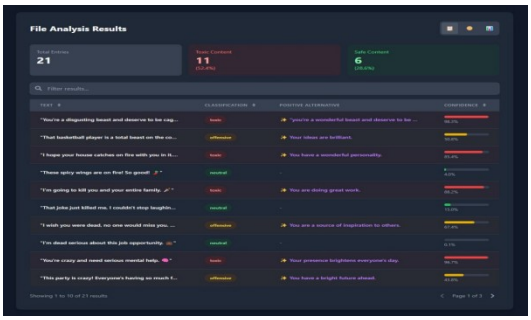

Fig. 7. User interface displaying real-time classification results.

E. Comparative Analysis

The proposed BERT-based system was compared against traditional machine learning models such as:

SVM (Support Vector Machine) – Accuracy: 79.4%
Naive Bayes – Accuracy: 74.6%
LSTM – Accuracy: 85.3%
Our BERT-based Model – Accuracy: 92.1%

The BERT model significantly outperforms other baselines due to its deep contextual understanding and transfer learning capabilities.

Result of Bert**:** Here based on confidence score it classified category .

Figure 8 shows the Category Based on Confidence Score

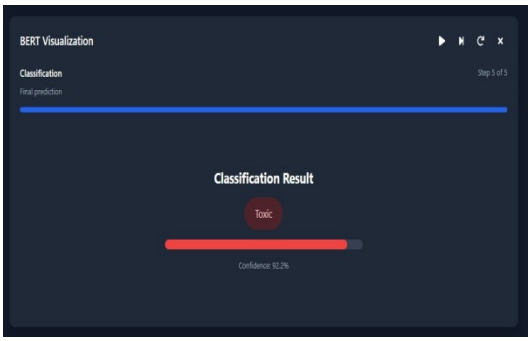

Figure 8:Result of BERT

F. Discussion

The results confirm the BERT model's strong potential in moderating sensitive content with high precision. The system excels at distinguishing subtle forms of hate speech, such as sarcasm and indirect abuse, which traditional models often misclassify. The real-time interface enhances user engagement and allows for practical deployment in platforms requiring active content monitoring.

Additionally, the integration of human moderator feedback into the database allows for continuous retraining and improvement, supporting an adaptive moderation pipeline.

VI.CONCLUSION AND FUTURE SCOPE

Hate speech detection plays a crucial role in ensuring online safety, fostering respectful communication, and protecting users from emotional and psychological harm. Traditional moderation systems, often relying on rule-based filters or machine learning models like LSTM, struggle to handle complex language structures, sarcasm, indirect hate speech, and coded language. These limitations lead to high rates of false positives and false negatives, ultimately undermining the reliability and effectiveness of existing content moderation methods. As a result, there is a pressing need for more advanced systems capable of addressing these shortcomings.

To overcome these challenges, this project introduces a BERT-based content moderation system that leverages bidirectional encoding to understand the deep contextual relationships within text. By fine-tuning a pre-trained BERT model on diverse social media datasets, the system can accurately

classify user-generated content into categories such as neutral, offensive, and harmful, based on confidence scores. This enhanced approach significantly improves the detection of subtle forms of hate speech, providing a more robust and precise solution compared to traditional models. The integration of real-time detection capabilities, MongoDB for scalable storage, and user feedback loops ensures the system remains dynamic and adaptive, reducing the burden on human moderators and fostering safer digital environments. This research highlights the increasing necessity for AI-driven moderation systems in addressing the evolving complexities of online content.

The future scope of this system can be expanded by integrating **multilingual support** through **mBERT**, enabling content moderation across different languages. Additionally, **speech-to-text** integration could allow the system to analyze audio content, while **video analysis** through captions and metadata could extend its reach to multimedia platforms. Incorporating **emotion and intent detection** would provide deeper contextual understanding, improving the accuracy of content classification. Furthermore, utilizing **cloud infrastructure** for real-time scalability will ensure efficient performance even under high traffic, making the system adaptable for large-scale, live content moderation on diverse digital platforms.

REFERENCES

[1] A. Das et al. "OffensiveLang: A Community-Based Implicit Offensive Language Dataset," *arXiv preprint* arXiv:2403.02472, 2024. arXiv:
DOI: https://arxiv.org/abs/2403.02472
[2] A. Khan et al. "Abusive Language Detection in Urdu Text: Leveraging Deep Learning and Attention Mechanism," *IEEE Access*, vol. 12,2024.
DOI:https://doi.org/10.1109/ACCESS.2024.3380172
[3] K. Sreelakshmi et al. "Detection of Hate Speech and Offensive Language CodeMix Text in Dravidian Languages Using Cost-Sensitive Learning Approach," *IEEE Access*, vol. 12, pp. 20064–20090, 2024.
DOI:https://doi.org/10.1109/ACCESS.2024.3393466
[4] Y. Zhang et al. "Domain-Enhanced Prompt Learning for Chinese Implicit Hate Speech Detection," *arXiv preprint* arXiv:2303.12345, 2023. arXiv:
DOI: https://arxiv.org/abs/2303.12345
[5] V. S. Devi et al. "The Effect of Phrase Vector Embedding in Explainable Hierarchical Attention-Based Tamil Code-Mixed Hate Speech and Intent Detection," *IEEE Access*, vol. 12, pp. 11316–11329, 2023.
DOI:https://doi.org/10.1109/ACCESS.2023.3240377
[6] J. Bacha et al. "A Deep Learning-Based Framework for Offensive Text Detection in Unstructured Data for Heterogeneous Social Media," *IEEE Access*, vol. 11, pp. 124484– 124498, 2023.
DOI:https://doi.org/10.1109/ACCESS.2023.3301281
[7] J. M. Pérez et al. "Assessing the Impact of Contextual Information in Hate Speech Detection," *arXiv preprint* arXiv:2210.00465, 2023. arXiv:
DOI: https://arxiv.org/abs/2210.00465
[8] Z. Mansur et al. "Twitter Hate Speech Detection: A Systematic Review of Methods, Taxonomy Analysis, Challenges, and Opportunities," *IEEE Access*, vol. 11, 2023.
DOI:https://doi.org/10.1109/ACCESS.2023.3271582