International Journal for Modern Trends in Science and Technology
Volume 12, Issue 04, pages 24-29.
ISSN: 2455-3778 online
Available online at: http://www.ijmtst.com/vol12issue04.html
DOI: https://doi.org/10.5281/zenodo.19321919

Detecting Online Abuse and Aggressive Behavior in British English using Machine Learning

Guna Gayathri Praseetha K, D Srilekha

Department of Computer Science and Engineering, PBR Visvodaya Institute of Technology and Science, Kavali, Andhra Pradesh, India.

To Cite this Article
Guna Gayathri Praseetha K & D Srilekha (2026). Detecting Online Abuse and Aggressive Behavior in British English using Machine Learning. International Journal for Modern Trends in Science and Technology, 12(04), 24-29. https://doi.org/10.5281/zenodo.19321919

Article Info
Received: 28 February 2026; Revised: 18 March 2026; Accepted: 22 March 2026.

KEYWORDS

Abusive Language Detection, Machine Learning, Natural Language Processing (NLP), Text Classification, Sentiment Analysis, Automated Content Moderation.

ABSTRACT

The rapid expansion of digital communication platforms has increased the prevalence of abusive and aggressive language online, creating challenges for maintaining safe and respectful virtual environments. This study explores the use of machine learning techniques to detect abusive and aggressive behavior in British English within user-generated content. Annotated datasets containing abusive and non-abusive text are used to train classification models. Linguistic and contextual features—such as offensive words, hostile tone, and sentiment patterns—are analyzed to improve detection accuracy. The performance of the models is evaluated using metrics like accuracy, precision, recall, and F1-score. The results show that combining linguistic and contextual features improves the identification of abusive language. Overall, the study highlights the potential of machine learning to support automated moderation systems and enhance online safety.

CYBER SECURITY

Detecting online abuse and aggressive behavior in British English using machine learning is a vital aspect of cyber security, as harmful communication threatens the safety of digital platforms. Machine learning models trained on annotated datasets can identify abusive text by analyzing linguistic signals such as offensive words, hostile tone, and sentiment patterns. This strengthens automated moderation systems, which act as a defense layer against cyber threats like harassment, misinformation, and toxic interactions. By applying evaluation metrics such as accuracy, precision, recall, and F1-score, these systems ensure reliable detection and reduce risks of false classification. A practical example is the filtering of abusive comments on social media, which helps protect users from psychological harm and maintains secure online environments.

OBJECTIVE

The rapid growth of digital communication platforms has increased the spread of abusive and aggressive

language online. This creates challenges in maintaining safe and respectful virtual environments. Manual moderation is often insufficient due to the large volume of user-generated content, so machine learning techniques are used to detect abusive behaviour automatically. Annotated datasets containing abusive and non-abusive text help train classification models to recognize harmful language patterns. Linguistic and contextual features such as offensive words, hostile tone, and sentiment are analyzed to improve detection accuracy. The performance of these models is evaluated using metrics like accuracy, precision, recall, and F1-score. Overall, machine learning plays an important role in supporting automated moderation systems and improving safety and trust in online communities.

NEED FOR STUDY

With the rapid growth of digital communication platforms, the spread of abusive and aggressive language in online content has increased significantly. This creates challenges in maintaining safe and respectful digital environments. Therefore, there is a need for automated systems that can detect harmful communication in British English. Machine learning techniques can analyze large amounts of user- generated text and identify abusive patterns using linguistic and contextual features such as offensive words, hostile tone, and sentiment. By training models with annotated datasets and evaluating them using metrics like accuracy, precision, recall, and F1-score, effective detection systems can be developed to support online moderation and improve safety in digital platforms.

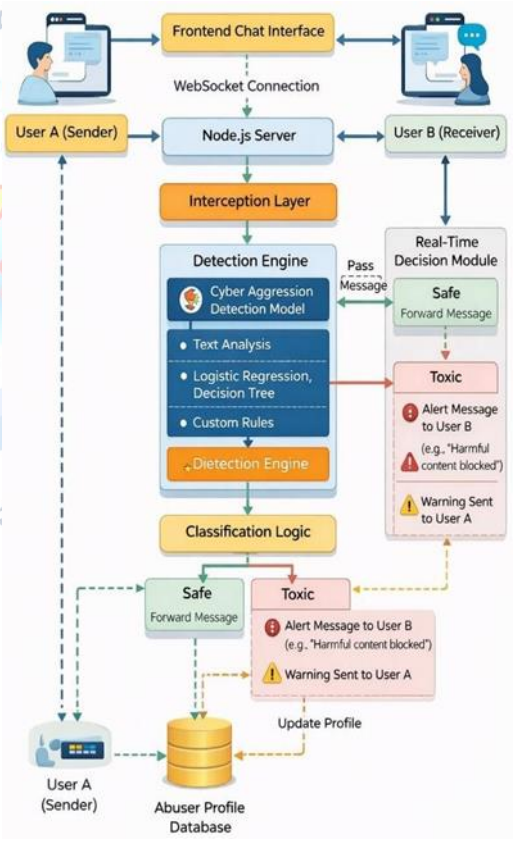
EXISTING SYSTEM

Existing systems for detecting online abuse and aggressive behavior in British English primarily focus on text-only classification, with most studies formulating the task as binary classification (abusive vs. non-abusive) rather than detailed multi-class categorization such as insults, threats, harassment, or hate speech. Early approaches relied on traditional machine learning algorithms such as Naive Bayes (NB), Support Vector Machine (SVM), Logistic Regression, Random Forest (RF), and AdaBoost, combined with feature extraction techniques like Bag of Words (BoW), TF-IDF, and word or character n-grams. Although these methods achieved moderate performance, they were limited in capturing contextual meaning, sarcasm, implicit aggression, and British slang or informal expressions commonly used in online platforms.

Disadvantages

- Mostly works as binary classification (abusive or non-abusive) and does not identify detailed categories like threats or harassment.
- Uses text-only data and ignores user behavior or conversation context.
- Cannot properly understand context and full meaning of sentences.
- Fails to detect sarcasm or indirect aggression.
- Struggles with British slang, short forms, and informal language.

SYSTEM ARCHITECTURE



SYSTEM REQUIREMENTS

Hardware Requirements

Processor - i7-1255U (1.70 GHz)

RAM - 16.0 GB

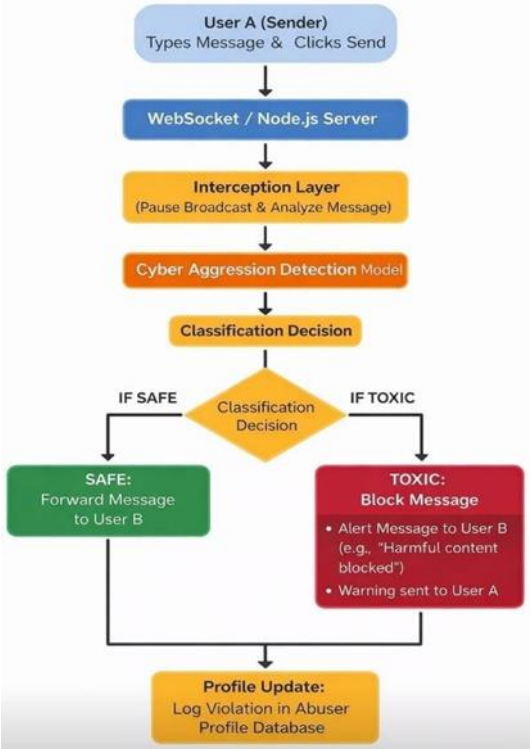
HDD - 500 GB

Software Requirements

Operating System - Window 10 or above Programming Language - Python

Front End - HTML, CSS
Back End - MySQL
Tool- Xampp

Modules Description



• User Module

The User Module enables real-time communication between the sender and the recipient through a chat interface. It allows users to compose, send, and receive messages seamlessly. This module also provides instant feedback in the form of warnings or safety alerts when harmful content is detected.

• Connection and Interception Module

The Connection and Interception Module manage real-time message transmission using WebSocket technology. It intercepts outgoing messages before they are broadcast to the recipient. This temporary pause allows the system to forward the message to a server-side middleware for analysis.

• Detection Engine

The Detection Engine is responsible for analyzing message content using a specialized NLP-based pipeline. It examines text for signs of cyber aggression, toxicity, and regional slang usage.

Based on the analysis, the engine classifies messages as either safe or toxic.

• Behavioral Profiling Module

The Behavioral Profiling Module maintains a record of user behavior by logging detected violations in a database. Each toxic interaction contributes to updating the user's aggression score. When predefined toxicity thresholds are exceeded, the system can automatically restrict or monitor the user's account.

Challenges & Risks

• Technical Challenges (The "Tech" Problems):

• Slow Speeds: Checking every message with AI can cause a delay or lag, making the chat feel slow instead of instant.

• Understanding Jokes: The AI might struggle to tell the difference between a mean insult and a joke between friends, leading to "false alarms."

• Changing Slang: People often use symbols or new slang to hide mean words, so the system must be constantly updated to stay smart.

• Ethical Concerns (The "People" Problems)

• Privacy: Because the system "reads" messages to keep people safe, some users might feel like they are being watched or spied on.

• Wrongful Blocking: The AI might accidentally block someone who isn't doing anything wrong just because they use strong language or a specific accent.

• Over-Censorship: There is a risk that the system becomes too strict, stopping people from having normal, heated arguments or expressing themselves freely.

PROPOSED SYSTEM

The proposed system focuses on detecting abusive and aggressive language in British English using machine learning techniques. The main idea of this system is to automatically analyse user-generated content from online platforms and classify whether the text is abusive or non-abusive. By applying machine learning algorithms and natural language processing (NLP) techniques, the system can identify offensive expressions, hostile tone, and negative sentiment patterns in messages. The system uses labelled datasets to train classification models and helps moderators detect harmful behaviour quickly and efficiently.

Advantages

• Automatically detects abusive and aggressive language in online content

• Improves safety and moderation in digital communication platforms
• Processes large volumes of text quickly and efficiently
• Reduces the need for manual content moderation
• Helps identify offensive words, hostile tone, and negative sentiment
• Supports online platforms in maintaining respectful communication

Hardware Requirements
• CPU type: Intel Pentium 4
• Clock speed: 3.0 GHz
• RAM size: 512 MB
• Hard disk capacity: 40 GB
• Monitor type: 15 Inch color monitor
• Keyboard type: Internet keyboard

Software Requirements
• Operating System: Windows OS
• Language: Python / PHP
• Back End: MySQL
• IDE: NetBeans / Jupyter Notebook

TECHNIQUES USED IN THE PROJECT VERIFIABLE OUTSOURCED DECRYPTION

Setup ()

1. Data Collection ()

Let the dataset be represented as:

$D = \{(t1, y1), (t2, y2), (t3, y3), \ldots, (tn, yn)\}$

Where:
• tirepresents the text message or comment collected from online platforms.
• yirepresents the label of the text.

$yi \in \{0,1\}$
• 0→ Non-abusive text
• 1→ Abusive text

Explanation:

In this step, a dataset of online messages is collected from sources like social media or forums. Each message is labelled as abusive or non-abusive. This labelled data is used to train the machine learning model.

2. Preprocessing ()

Each text message is cleaned using a preprocessing function:

$t' = Clean(ti)$

Where the Clean () function performs:
• Converting text to lowercase
• Removing punctuation and symbols
• Removing stop words (such as the, is, a)
• Tokenizing the sentence into words

Explanation:

Raw text data usually contains unnecessary words and symbols. Preprocessing removes these unwanted elements so that the machine learning model can focus on meaningful words for analysis.

3. Feature Extraction ()

Each processed text is converted into a numerical feature vector:

$Xi = (x1, x2, x3, \ldots, xm)$

Where features may include:
• Frequency of offensive words
• Sentiment score of the sentence
• Contextual language patterns

Explanation:

Machine learning models cannot understand raw text directly. Therefore, the text is converted into numerical values called features, which represent important characteristics of the message.

4. Model Training ()

A machine learning classifier is trained using the dataset:

$f(X) = y$

through preprocessing and feature extraction. The trained model then predicts whether the message contains abusive language.

Where:

i i

detected as abusive)
• FN = False Negative (abusive text not

• Xi= Feature vector
• yi= Label (abusive or non-abusive)
• f= Classification model Examples of models used:
• Naïve Bayes
• Support Vector Machine (SVM)
• Logistic Regression

Explanation:

During training, the algorithm learns patterns from the dataset and builds a model that can differentiate abusive language from normal text.

5. Classification ()

For a new input message t:

$X = Feature(t)$

The trained model predicts:

$\hat{y} = f(X)$

Where:

- $\hat{y} = 1 \rightarrow$ Abusive text
- $\hat{y} = 0 \rightarrow$ Non-abusive text

Explanation:

When a new message is entered into the system, it goes through preprocessing and feature extraction. The trained model then predicts whether the message contains abusive language.

6. Model Evaluation ()

The performance of the model is measured using the following formulas:

Accuracy

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision

$$Precision = \frac{TP}{TP + FP}$$

Recall

$$Recall = \frac{TP}{TP + FN}$$

F1 Score

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Where:

- TP = True Positive (correctly detected abusive text)
- TN = True Negative (correctly detected non- abusive text)
- FP = False Positive (normal text incorrectly
- FN = False Negative (abusive text not detected)

Explanation:

These evaluation metrics help measure how accurately the system detects abusive and aggressive language in online communication. Higher values indicate better performance of the detection model.

CONCLUSION

This study presents a machine learning based framework for detecting abusive and aggressive behaviour in British English within online communication platforms. With the rapid increase in user-generated content, maintaining safe and respectful digital environments has become a major challenge. The proposed system analyses textual data using natural language processing techniques and machine learning algorithms to identify harmful language patterns. By using annotated datasets containing abusive and non-abusive text, the system learns to recognize offensive expressions, hostile tone, and negative sentiment in online messages.

The study demonstrates that combining linguistic and contextual features significantly improves the accuracy of abusive language detection. The performance of the models is evaluated using metrics such as accuracy, precision, recall, and F1-score, which indicate the effectiveness of the system in identifying aggressive content. The developed approach supports automated moderation systems that can assist online platforms in reducing harmful communication. Overall, the proposed framework contributes to improving safety, trust, and responsible interaction in digital communities by enabling efficient detection of abusive language.

FUTURE ENHANCEMENT

In the future, the system can be improved by using advanced deep learning models and larger datasets to increase detection accuracy. The model can also be extended to support multiple languages and detect complex forms of abusive language such as sarcasm and hidden aggression. Additionally, real-time monitoring systems can be developed to automatically detect and filter abusive content on online platforms, helping to create safer digital communication environments.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," in Proc. 11th Int. AAAI Conf. Web Social Media, 2017, pp. 512–515.

[2] Z. Waseem and D. Hovy, "Hateful symbols or hateful people? Predictive features for hate speech detection on Twitter," in Proc. NAACL Student Research Workshop, 2016, pp. 88–93.

[3] P. Fortuna and S. Nunes, "A survey on automatic detection of hate speech in text," ACM Computing Surveys, vol. 51, no. 4, pp. 1–30, 2018.
[4] T. Badjatiya, S. Gupta, M. Gupta, and V. Varma, "Deep learning for hate speech detection in tweets," in Proc. 26th Int. World Wide Web Conf. Companion, 2017, pp. 759–760.
[5] I. Kwok and Y. Wang, "Locate the hate: Detecting tweets against blacks," in Proc. AAAI Conf. Artificial Intelligence, 2013, pp. 1621–1622.
[6] H. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang, "Abusive language detection in online user content," in Proc. 25th Int. World Wide Web Conf., 2016, pp. 145–153.
[7] E. Dinakar, R. Reichart, and H. Lieberman, "Modeling the detection of textual cyberbullying," in Proc. Int. AAAI Conf. Web Social Media, 2011, pp. 11–17.
[8] S. Malmasi and M. Zampieri, "Detecting hate speech in social media," in Proc. Int. Conf. Recent Advances in Natural Language Processing, 2017, pp. 467–472.
[9] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar, "Predicting the type and target of offensive posts in social media," in Proc. NAACL-HLT, 2019, pp. 1415–1420.
[10] A. Schmidt and M. Wiegand, "A survey on hate speech detection using natural language processing," in Proc. 5th Int. Workshop on Natural Language Processing for Social Media, 2017, pp. 1–10.
[11] S. Burnap and M. L. Williams, "Cyber hate speech on Twitter: An application of machine classification and statistical modeling," Information & Management, vol. 54, no. 2, pp. 273–282, 2017.
[12] J. Gao and H. Huang, "Detecting online hate speech using context-aware models," in Proc. IEEE Int. Conf. Data Mining Workshops, 2017, pp. 260– 265.
[13] Y. Zhang, B. Robinson, and M. Tepper, "Detecting hate speech on Twitter using deep learning," Semantic Web Journal, vol. 10, no. 5, pp. 1–12, 2019.
[14] P. Mishra, M. Del Tredici, H. Yannakoudakis, and E. Shutova, "Abusive language detection with graph convolutional networks," in Proc. ACL, 2019, pp. 2736–2745.
[15] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," in Proc. European Chapter of the Association for Computational Linguistics, 2017, pp. 427–431.
[16] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT, 2019, pp. 4171–4186.
[17] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in Proc. Int. Conf. Learning Representations, 2013.
[18] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.
[19] C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.
[20] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in Proc. European Conf. Machine Learning, 1998, pp. 137–142.