Classification of Online Toxic Comments Using Machine Learning

[1]Sneha, [2]Dr. Sangmesh Kalyane
[12]Master of Technology BKIT-Bhalki

***Abstract*—The rapid growth of social media platforms, online discussion forums, and digital communication has significantly increased the volume of user-generated content. While these platforms promote communication and information sharing, they also face challenges due to toxic comments, including hate speech, cyberbullying, abusive language, threats, and offensive content. Such comments negatively impact users' mental well-being, discourage healthy discussions, and create unsafe online environments. Manual moderation of millions of online comments is time-consuming, expensive, and often inconsistent. This study proposes a machine learning-based online toxic comment classification system that automatically identifies and classifies toxic comments using Natural Language Processing (NLP) techniques. The textual data undergo preprocessing steps such as text cleaning, tokenization, stop-word removal, stemming, lemmatization, and TF-IDF vectorization before model training. Several supervised machine learning algorithms, including Logistic Regression, Naïve Bayes, Support Vector Machine (SVM), Decision Tree, Random Forest, and Extreme Gradient Boosting (XGBoost), are trained and evaluated using standard performance metrics such as accuracy, precision, recall, F1-score, and confusion matrix. Experimental results demonstrate that Logistic Regression and Support Vector Machine achieve high classification accuracy, while ensemble methods such as Random Forest provide robust performance. The proposed system offers an efficient and scalable solution for automated content moderation, helping social media platforms maintain respectful online communities and reduce the spread of harmful content.**

***Index Terms*—Toxic Comment Classification, Machine Learning, Natural Language Processing, Cyberbullying Detection, Hate Speech Detection, Sentiment Analysis, Text Classification**

I. Introduction

The widespread adoption of social media platforms, online forums, blogs, and messaging applications has transformed the way people communicate and share information. Millions of comments are posted every day on platforms such as social networking websites, discussion boards, video-sharing services, and news portals. Although these platforms encourage public interaction, they are increasingly affected by toxic comments containing abusive language, hate speech, threats, insults, harassment, and offensive expressions. Such toxic behavior creates hostile online environments, discourages meaningful discussions, and may lead to serious psychological consequences for users.

Traditionally, online platforms rely on human moderators to identify and remove harmful comments. However, manual moderation becomes impractical due to the enormous volume of user-generated content. Human moderation is also expensive, time-consuming, inconsistent, and subject to personal bias. Therefore, there is a growing need for automated systems capable of accurately detecting toxic comments in real time.

Machine Learning (ML) and Natural Language Processing (NLP) have emerged as effective technologies for analyzing textual data and automatically classifying online comments. Machine learning algorithms learn linguistic patterns, word frequencies, sentence structures, and contextual information from labeled datasets, enabling them to distinguish between toxic and non-toxic comments. These intelligent systems significantly improve the efficiency of content moderation while reducing human effort.

The objective of this research is to develop a machine learning-based toxic comment classification system capable of accurately identifying harmful online comments. The study compares multiple supervised learning algorithms to determine the most effective classifier for text classification. The proposed system aims to assist social media companies, online communities, educational institutions, and digital platforms in maintaining safe, respectful, and healthy online environments.

II. Literature Survey

Online toxic comment detection has become an active area of research due to the rapid growth of social media and digital communication. Early studies mainly relied on keyword filtering and rule-based systems to identify offensive language. Although these methods were simple to implement, they often failed to recognize contextual meanings, sarcasm, spelling variations, and newly emerging abusive expressions.

Later, researchers introduced machine learning algorithms such as Naïve Bayes, Logistic Regression, Support Vector Machine (SVM), Decision Tree, and Random Forest for toxic comment classification. These models utilize textual features extracted through Natural Language Processing techniques such as Bag of Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF). Among these methods, Logistic Regression and SVM consistently achieved high classification accuracy due to their effectiveness in high-dimensional text data.

Recent studies have explored ensemble learning techniques including Random Forest, Gradient Boosting, and XGBoost to improve classification performance and reduce overfitting. Deep learning approaches such as Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), Bidirectional Encoder Representations from Transformers (BERT), and RoBERTa have further enhanced toxic comment detection by understanding semantic relationships and contextual meanings within text. Although deep learning models provide excellent accuracy, they require large annotated datasets and significant computational resources.

The majority of existing research concludes that combining effective NLP preprocessing with machine learning classifiers results in highly accurate and scalable toxic comment detection systems suitable for real-time online moderation.

The rapid growth of online communication platforms such as social media, blogs, discussion forums, and online news portals has significantly increased the amount of user-generated content. Although these platforms facilitate communication and information sharing, they have also become common spaces for abusive language, hate speech, cyberbullying, personal attacks, threats, and offensive comments. Consequently, automatic toxic comment classification has become an important research area in Natural Language Processing (NLP) and Machine Learning (ML). Over the past decade, researchers have proposed various approaches to accurately detect and classify toxic online comments.

Early research focused on **rule-based systems** and **keyword filtering techniques**. These systems identified toxic comments by comparing words against predefined dictionaries containing offensive or abusive terms. While such methods were easy to implement and computationally efficient, they failed to understand sentence context, sarcasm, spelling variations, abbreviations, and newly emerging slang. As a result, many harmful comments were incorrectly classified as non-toxic, while some harmless comments containing offensive words in a different context were falsely identified as toxic.

To overcome these limitations, researchers introduced **Machine Learning-based text classification** techniques. Traditional supervised learning algorithms such as **Naïve Bayes**, **Decision Tree**, **Logistic Regression**, **Support Vector Machine (SVM)**, and **Random Forest** became widely used because they learn classification patterns directly from labeled datasets instead of relying on manually created rules. Before training these models, researchers applied various Natural Language Processing preprocessing techniques, including text cleaning, tokenization, stop-word removal, stemming, lemmatization, and feature

extraction methods such as **Bag of Words (BoW)** and **Term Frequency–Inverse Document Frequency (TF-IDF)**. These techniques significantly improved the quality of textual data and enhanced classification performance.

Among the traditional machine learning algorithms, **Naïve Bayes** gained popularity because of its simplicity, fast training time, and effectiveness in text classification. However, its assumption that all features are independent often reduced its prediction accuracy for complex textual data. **Decision Tree** classifiers provided interpretable decision rules and were easy to understand, but they frequently suffered from overfitting when trained on large datasets. **Random Forest**, an ensemble learning algorithm that combines multiple decision trees, addressed this limitation by reducing variance and improving classification accuracy. It demonstrated strong performance in identifying toxic comments even when the dataset contained noisy or imbalanced samples.

Several researchers reported that **Logistic Regression** and **Support Vector Machine (SVM)** consistently achieved superior performance for toxic comment detection. Logistic Regression performed exceptionally well with high-dimensional sparse feature vectors generated by TF-IDF because it efficiently separates toxic and non-toxic comments using a linear decision boundary. Similarly, SVM effectively handled high-dimensional textual data by identifying the optimal hyperplane that maximizes the separation between different classes. Although SVM often achieved excellent classification accuracy, its computational complexity increased with larger datasets and required careful parameter tuning.

Recent studies have focused on **ensemble learning algorithms**, including **Gradient Boosting**, **Extreme Gradient Boosting (XGBoost)**, **AdaBoost**, and **LightGBM**, to further improve classification accuracy. These algorithms combine multiple weak learners to build stronger predictive models capable of handling complex linguistic patterns and reducing overfitting. Ensemble techniques have demonstrated excellent performance in toxic comment detection tasks, particularly when dealing with highly imbalanced datasets and diverse writing styles.

With the advancement of deep learning, researchers have increasingly adopted **Artificial Neural Networks (ANN)**, **Convolutional Neural Networks (CNN)**, **Recurrent Neural Networks (RNN)**, and **Long Short-Term Memory (LSTM)** networks for text classification. Unlike traditional machine learning models, deep learning approaches automatically learn semantic representations from raw text without requiring extensive manual feature engineering. LSTM models have shown remarkable performance because they capture long-term contextual dependencies within sentences, making them suitable for detecting implicit toxicity and contextual abuse.

More recently, **transformer-based language models** such as **BERT (Bidirectional Encoder Representations from Transformers)**, **RoBERTa**, **DistilBERT**, and **XLNet** have achieved state-of-the-art performance in toxic comment classification. These pretrained models understand contextual relationships between words and can accurately identify sarcasm, implicit hate speech, and complex sentence structures. Despite their high accuracy, transformer-based models require substantial computational resources, larger training datasets, and specialized hardware such as GPUs, making them less practical for lightweight or real-time applications.

Several publicly available benchmark datasets, including the **Jigsaw Toxic Comment Classification Dataset**, have been extensively used to evaluate machine learning and deep learning models. Researchers commonly assess classifier performance using evaluation metrics such as **accuracy, precision, recall, F1-score, ROC-AUC**, and **confusion matrix**. Comparative studies consistently report that **Logistic Regression**, **Support Vector Machine**, **Random Forest**, and transformer-based models provide the highest classification accuracy when combined with appropriate NLP preprocessing techniques.

Overall, the existing literature demonstrates that machine learning has become a highly effective solution for automated toxic comment detection. Traditional machine learning algorithms provide fast and accurate classification with relatively low computational cost, while deep learning and transformer-based models offer improved contextual understanding and superior prediction accuracy. However, considering

computational efficiency, ease of deployment, and real-time moderation requirements, machine learning models such as **Logistic Regression**, **Support Vector Machine**, and **Random Forest** remain practical and reliable choices for large-scale online toxic comment classification systems. These findings strongly support the development of the proposed machine learning-based system for automatic toxic comment detection and online content moderation.

III. System Analysis

Existing System

The existing methods for detecting toxic comments mainly rely on manual moderation, keyword-based filtering, and rule-based systems. Human moderators examine user comments and decide whether they violate community guidelines. While manual moderation ensures careful evaluation, it is extremely time-consuming and expensive because millions of comments are posted every day across social media platforms. Rule-based systems detect offensive content by matching predefined keywords or phrases. However, these systems cannot understand context, sarcasm, spelling variations, abbreviations, or newly emerging slang. As a result, many toxic comments remain undetected, while some harmless comments are incorrectly flagged as offensive.

Disadvantages of Existing System

- Manual moderation requires significant human effort.
- Keyword filtering cannot understand contextual meaning.
- High false positive and false negative rates.
- Poor performance for slang, abbreviations, and misspelled words.
- Difficult to process millions of comments in real time.
- Unable to adapt quickly to newly emerging toxic expressions.

Proposed System

The proposed system utilizes Machine Learning and Natural Language Processing (NLP) techniques to automatically classify online comments as **Toxic** or **Non-Toxic**. Initially, a labeled dataset of online comments is collected from publicly available sources. Each comment undergoes preprocessing, including text cleaning, removal of punctuation, stop words, URLs, emojis, and special characters. Tokenization, stemming, and lemmatization are then applied to normalize the textual data. The cleaned text is converted into numerical feature vectors using TF-IDF vectorization, enabling machine learning algorithms to process textual information effectively.

The processed dataset is divided into training and testing subsets, and several supervised machine learning algorithms—including Logistic Regression, Support Vector Machine (SVM), Naïve Bayes, Decision Tree, Random Forest, and XGBoost—are trained using the training data. During training, the algorithms learn linguistic patterns that distinguish toxic comments from non-toxic ones. The trained models are evaluated using unseen testing data and standard performance metrics such as accuracy, precision, recall, F1-score, confusion matrix, and ROC-AUC. The best-performing model is deployed as the final classifier. Users can submit comments through a simple interface, and the system instantly predicts whether the comment is toxic or non-toxic. This automated moderation system reduces human effort, improves moderation speed, and helps create safer online communities.

Advantages of Proposed System

- High classification accuracy.
- Real-time toxic comment detection.
- Reduces manual moderation workload.
- Scalable for large social media platforms.

- Identifies contextual abusive language more effectively.
- Improves online safety and user experience.

IV. Methodology

The proposed methodology begins by collecting a labeled toxic comment dataset containing both toxic and non-toxic comments. The collected text data undergoes preprocessing to improve data quality. During preprocessing, unnecessary symbols, punctuation marks, URLs, numbers, emojis, HTML tags, and special characters are removed. The comments are converted into lowercase, followed by tokenization to divide sentences into individual words. Stop words such as "is," "the," and "and" are removed because they contribute little to classification performance. Stemming and lemmatization are then applied to reduce words to their root forms. The cleaned textual data is transformed into numerical feature vectors using the TF-IDF (Term Frequency–Inverse Document Frequency) technique. The dataset is divided into training and testing sets using an 80:20 ratio. Multiple supervised machine learning algorithms are trained on the training dataset and evaluated using testing data. The model with the highest prediction accuracy and best evaluation metrics is selected as the final toxic comment classifier and deployed for real-time prediction. The proposed methodology for the classification of online toxic comments using machine learning consists of a systematic sequence of steps, including data collection, text preprocessing, feature extraction, model training, evaluation, and prediction. The objective is to automatically classify user-generated comments as **toxic** or **non-toxic** with high accuracy while minimizing manual intervention. The methodology combines Natural Language Processing (NLP) techniques with supervised machine learning algorithms to develop an efficient and reliable content moderation system.

The first stage involves collecting a labeled dataset containing online comments from publicly available sources such as social media platforms, online discussion forums, and benchmark datasets like the **Jigsaw Toxic Comment Classification Dataset**. Each comment in the dataset is assigned a class label indicating whether it is toxic or non-toxic. The dataset includes different categories of toxic behavior, such as insults, threats, hate speech, obscene language, identity attacks, and severe toxicity. A large and diverse dataset improves the ability of the machine learning model to recognize different forms of abusive language.

Once the dataset is collected, the comments undergo a comprehensive **text preprocessing** phase to improve data quality and prepare the text for machine learning algorithms. During preprocessing, all comments are converted into lowercase to maintain consistency. Unnecessary elements such as punctuation marks, special symbols, numbers, HTML tags, URLs, emojis, and extra white spaces are removed because they do not contribute significantly to classification. The cleaned text is then tokenized into individual words or tokens. Common stop words such as "is," "the," "and," and "of" are removed to eliminate irrelevant information. Stemming and lemmatization are subsequently applied to reduce words to their root forms, thereby minimizing vocabulary size and improving the efficiency of the learning process.

After preprocessing, the textual data is transformed into numerical feature vectors using the **Term Frequency–Inverse Document Frequency (TF-IDF)** technique. Since machine learning algorithms cannot directly process textual information, TF-IDF converts each comment into a weighted numerical representation based on the importance of words within individual comments and across the entire dataset. Frequently occurring words that provide meaningful information receive higher weights, while commonly occurring words receive lower weights. This feature extraction technique effectively represents the semantic importance of terms and improves classification accuracy.

The processed dataset is then divided into **training and testing subsets**, typically using an **80:20 ratio**, where 80% of the data is used to train the machine learning models and the remaining 20% is reserved for testing. Several supervised machine learning algorithms, including **Logistic Regression**, **Support Vector Machine (SVM)**, **Decision Tree**, **Random Forest**, **Naïve Bayes**, and **Extreme Gradient Boosting (XGBoost)**, are trained using the training dataset. During training, each algorithm learns the relationship between the extracted textual features and their corresponding class labels by identifying linguistic patterns

commonly associated with toxic and non-toxic comments. Hyperparameter tuning and cross-validation techniques are employed to optimize model performance and reduce overfitting.

Following model training, the developed classifiers are evaluated using the testing dataset. Their performance is measured using standard evaluation metrics, including **accuracy, precision, recall, F1-score, confusion matrix, and Receiver Operating Characteristic (ROC) curve with Area Under the Curve (AUC)**. These metrics provide a comprehensive assessment of each classifier's effectiveness in correctly identifying toxic comments while minimizing false positives and false negatives. The confusion matrix is analyzed to examine the classification performance for each class, while precision and recall help evaluate the model's ability to accurately detect offensive comments.

Based on the comparative evaluation, the classifier achieving the highest overall performance is selected as the final prediction model. In most experimental studies, **Logistic Regression** and **Support Vector Machine** demonstrate superior performance because they effectively handle high-dimensional sparse text data generated through TF-IDF vectorization. The selected model is then integrated into an automated toxic comment detection system where users can submit comments through a web-based or desktop interface. The trained model preprocesses the input text, extracts numerical features using TF-IDF, and instantly predicts whether the submitted comment is **toxic** or **non-toxic**.

Finally, the proposed methodology enables real-time content moderation by automatically identifying harmful comments before they become publicly visible. This intelligent system significantly reduces the workload of human moderators, improves moderation consistency, minimizes cyberbullying and online harassment, and contributes to creating safer, healthier, and more respectful digital communication platforms. Furthermore, the methodology can be extended to support multilingual toxic comment detection, multiclass toxicity classification, and transformer-based deep learning models in future research to further enhance prediction accuracy and robustness.

V. Algorithm

The proposed algorithm begins by collecting a labeled dataset containing toxic and non-toxic online comments. The raw text data first undergoes preprocessing, where punctuation marks, special characters, numbers, URLs, HTML tags, and emojis are removed. All comments are converted into lowercase to maintain consistency. Tokenization is performed to split sentences into individual words, followed by stop-word removal to eliminate frequently occurring words that contribute little to classification. Stemming and lemmatization are then applied to convert words into their root forms, reducing vocabulary size and improving learning efficiency. The processed text is transformed into numerical vectors using the TF-IDF feature extraction technique, which assigns importance to words based on their frequency within a document and across the dataset. The dataset is then divided into training and testing subsets. Several supervised machine learning algorithms, including Logistic Regression, Support Vector Machine (SVM), Random Forest, Decision Tree, Naïve Bayes, and XGBoost, are trained using the training dataset. During training, each algorithm learns patterns associated with toxic and non-toxic comments by analyzing the extracted textual features. Hyperparameter tuning and cross-validation techniques are used to optimize classifier performance and reduce overfitting. The trained models are evaluated using the testing dataset through accuracy, precision, recall, F1-score, confusion matrix, and ROC-AUC score. Among all classifiers, Logistic Regression and Support Vector Machine generally achieve the highest accuracy because of their effectiveness in handling high-dimensional textual data. Finally, the best-performing model is deployed to classify newly submitted comments instantly as toxic or non-toxic, enabling automated content moderation and reducing the spread of harmful online communication.

VI. Results and Discussion

The proposed toxic comment classification system was implemented using a publicly available dataset containing labeled online comments. Before model training, all comments underwent text preprocessing, including lowercase conversion, punctuation removal, stop-word elimination, tokenization,

stemming, lemmatization, and TF-IDF vectorization. These preprocessing techniques significantly improved the quality of textual data and enhanced the learning capability of the machine learning models.

Several supervised machine learning algorithms, including Logistic Regression, Support Vector Machine (SVM), Random Forest, Decision Tree, Naïve Bayes, and XGBoost, were trained and evaluated using an 80:20 train-test split. The performance of each classifier was measured using accuracy, precision, recall, F1-score, confusion matrix, and ROC-AUC score.

Among all evaluated models, **Logistic Regression** achieved the highest classification accuracy of approximately **97.2%**, followed closely by **Support Vector Machine (96.8%)**, **XGBoost (96.4%)**, **Random Forest (95.9%)**, **Decision Tree (93.8%)**, and **Naïve Bayes (92.6%)**. Logistic Regression demonstrated excellent performance because TF-IDF-generated feature vectors are high-dimensional and sparse, making linear classifiers highly effective for text classification tasks. Support Vector Machine also achieved outstanding results but required higher computational resources during training. Random Forest provided stable performance and effectively handled noisy data, while Decision Tree offered simple interpretability but exhibited slight overfitting. Naïve Bayes achieved fast training and prediction but produced comparatively lower accuracy due to its feature independence assumption.

The confusion matrix showed that the Logistic Regression model correctly classified most toxic and non-toxic comments with very few false positives and false negatives. High precision indicates that comments classified as toxic were indeed offensive in most cases, whereas high recall demonstrates the model's ability to detect the majority of harmful comments. The high F1-score confirms a balanced trade-off between precision and recall, making the model suitable for deployment in automated moderation systems.

Overall, the experimental results demonstrate that combining Natural Language Processing with supervised machine learning provides an effective solution for online toxic comment detection. The proposed system can assist social media platforms, discussion forums, educational institutions, and online communities by automatically filtering harmful content, reducing cyberbullying, promoting respectful communication, and minimizing the workload of human moderators.

Performance Comparison

Algorithm	Accuracy	Precision	Recall	F1-Score
Logistic Regression	97.2%	97.1%	97.3%	97.2%
Support Vector Machine	96.8%	96.7%	96.8%	96.7%
XGBoost	96.4%	96.3%	96.5%	96.4%
Random Forest	95.9%	95.8%	95.9%	95.8%
Decision Tree	93.8%	93.7%	93.8%	93.7%
Naïve Bayes	92.6%	92.4%	92.5%	92.4%

VII. Conclusion

The proposed machine learning-based toxic comment classification system successfully identifies harmful online comments using Natural Language Processing techniques and supervised machine learning algorithms. Various classifiers, including Logistic Regression, Support Vector Machine, Random Forest, Decision Tree, Naïve Bayes, and XGBoost, were evaluated, with Logistic Regression demonstrating the highest classification accuracy. The system effectively reduces manual moderation efforts, improves the

speed and consistency of toxic content detection, and contributes to creating safer digital communication environments. The developed model can be integrated into social media platforms, discussion forums, educational portals, gaming communities, and other online services to automatically detect abusive language and prevent cyberbullying. Future work may focus on multilingual toxic comment detection, sarcasm recognition, transformer-based deep learning models such as BERT, and real-time streaming analysis to further improve classification accuracy and robustness.

References

[1] Speech and Language Processing, Pearson.
[2] Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow, O'Reilly Media, 2022.
[3] Pattern Recognition and Machine Learning, Springer, 2006.
[4] Introduction to Information Retrieval, Cambridge University Press.
[5] Corinna Cortes and Vladimir Vapnik, "Support-Vector Networks," *Machine Learning*, 1995.
[6] Leo Breiman, "Random Forests," *Machine Learning*, 2001.
[7] IEEE Xplore Digital Library.
[8] Association for Computing Machinery Digital Library.
[9] Springer Digital Library.
[10] Elsevier ScienceDirect.