IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 11, November 2025

DOI: 10.17148/IJIREEICE.2025.131124

Toxic Comment Classification Using Ensemble Machine Learning Techniques

AATHITYA.A[1], KRISHITH TP[2], SARVESH S[3], KEVIN BENJAMIN SAMUEL[4], PRANAV M[5], VIGNESH D[6], Dr. M. ULAGAMMAI[7]

Department of Computer Science and Engineering (Emerging Technologies),

SRM Institute of Science and Technology, Vadapalani Campus, Chennai, 600026, India.

Abstract: The exponential growth of social media and online discussion forums increases the difficulty of maintaining healthy digital spaces with growing volumes of toxic comments. This study designs an efficient machine learning model that can classify toxic content by employing techniques in Natural Language Processing and ensemble learning. The approach mixes the models Logistic Regression, Random Forest, and XGBoost into a framework based on Voting Ensemble, boosting predictive accuracy. By using TF-IDF for feature extraction, along with a soft voting mechanism, the proposed ensemble outperforms the stand-alone classifiers in both ROC-AUC and precision. The system proposed here will provide a robust, efficient, and scalable way to identify and manage toxicity online.

Keywords: Toxic comments, Natural Language Processing, TF-IDF, Ensemble Learning, Voting Classifier, XGBoost, Logistic Regression, Random Forest.

I. INTRODUCTION

The internet revolutionized communications, opening an arena for discussion, learning, and collaboration at a global level. This openness, however, also promotes harmful language, including harassment and hate speech. The inefficiency of manual moderation motivates the development of automated systems. This paper will discuss an ensemble-based model with the inclusion of Logistic Regression, Random Forest, and XGBoost in identifying toxic

comments based on TF-IDF features. An ensemble approach makes the model more robust and outperforms the performance of single classifiers. The rapid proliferation of online platforms and social media has given birth to new ways to connect with others, share opinions, and gather information. This digital evolution has also brought forth new forms of toxic behaviors, such as hate speech, cyberbullying, harassment, and the spread of misinformation. Such toxic comments discourage user participation, damage their mental health, and deplete the quality of online discourse. Therefore, developing automatic systems that can detect toxic comments with high accuracy and filter them remains a very important challenge in modern NLP.

Toxic comment classification is a subcategory of text classification that involves detecting inappropriate, offensive, or harmful content within user-generated text. Unlike traditional sentiment analysis, mainly employed to grapple with positive or negative sentiments, toxicity detection has to tackle more complex linguistic patterns such as sarcasm, targeted insults, and context-contingent language. It is hence highly relevant to devise models which can grasp subtle differences between what is acceptable speech and what is not, in maintaining digital safety and inclusivity.

Recent breakthroughs in machine learning have advanced large-scale textual data analysis. Early research on toxicity detection was performed using classical algorithms like Logistic Regression and Support Vector Machines because these models are simple and interpretable. However, high dimensionality and non-linear relationships tied within text data are common challenges to these models. Ensemble models overcome such limitations by aggregating the strengths of several learners.

We focus on the detection of different forms of online toxicity, including obscene, threatening, insulting, and identity-based hate comments. In this work, we adopt an ensemble approach that combines three machine learning algorithms: Logistic Regression, Random Forest, and XGBoost. Whereas Logistic Regression represents a strong linear baseline, Random Forest is able to deal with high-dimensional feature spaces and complex decision boundaries, while XGBoost specifically targets performance refinements by incorporating gradient boosting. These models have been combined into a soft voting classifier, whose output probabilities have been averaged in order to obtain a balanced and accurate prediction over multiple toxicity categories.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 11, November 2025

DOI: 10.17148/IJIREEICE.2025.131124

The data have been taken from the Jigsaw Toxic Comment Classification Challenge and include over 150,000 labeled comments from Wikipedia discussion pages. Each comment has one or more associated labels of different toxicity types. This dataset will present a very strong benchmark for testing text classification models due to the diversity of linguistic patterns and real-world contexts.

Feature extraction is the most critical stage in the transformation of raw textual data into machine-understandable form. Text has to be changed into numerical feature vectors by the TF-IDF method, capturing the importance of each word in context. Both word-level and character-level n-grams have been utilized in capturing variations in text, spelling errors, and partial matches of abusive words.

The key objective of this work is to develop a machine learning pipeline that can offer a good generalization for different forms of toxic languages computationally efficiently. In this regard, the proposed ensemble system combines several algorithms to overcome the weakness of individual models. This would ensure higher robustness of the solution towards linguistic variations and noise. All the performance metrics such as accuracy, precision, recall, and ROC-AUC have been computed for the test dataset, showing the reliability and scalability of the model in practical applications.

In this light, this paper proposes a lightweight yet effective ensemble framework for the automatic classification of toxic comments. The proposed system can be deployed for moderators, social media platforms, and online communities to identify and filter harmful contents with the goal of achieving healthier digital interactions and safer online environments.

II. LITERATURE SURVEY

Previous works were focused on machine learning and deep learning approaches to detect toxicity. Traditional models used TF-IDF and linear classifiers such as Logistic Regression and SVM. More recently, deep learning models like BERT and LSTM were introduced to

While deep learning models lead on contextual understanding, ensemble models balance accuracy with computational efficiency by making use of combined strengths from different algorithms. It has been pointed out that model diversity enables ensembles to improve performance on complex datasets like Jigsaw.

III. METHODOLOGY

It follows the following stages: data collection, preprocessing, feature extraction, model training, and ensemble integration. In this work, the Jigsaw Toxic Comment dataset is utilized, which has six categories of toxicity. Preprocessing is comprised of text cleaning and token normalization. Text is transformed to feature vectors using TF-IDF vectorization. Training different base classifiers (Logistic Regression, Random Forest, XGBoost) whose outputs are combined together with the help of a soft voting ensemble mechanism allows for improved stability in classification.

One of the first large-scale toxic comment datasets was introduced by the Jigsaw and Conversation AI teams at Google through the Toxic Comment Classification Challenge in 2017, hosted on Kaggle. The benchmark dataset contained labeled comments from Wikipedia talk pages classified according to several types of toxicity, such as "toxic", "severe toxic", "obscene", "threat", "insult", and "identity hate". This competition triggered a number of researchers to explore several algorithms and feature engineering methods to enhance the performance of classification accuracy and the generalization of models.

Traditional machine learning algorithms include Naïve Bayes, Logistic Regression, and Random Forest, showing good performance on toxicity detection tasks, normally in combination with TF-IDF or word count vectorization. However, most of these models fail to capture deeper semantic and contextual relationships between words, which are extremely important for identifying complex linguistic cues such as sarcasm or implicit hate. Because of the mentioned shortcomings, more sophisticated algorithms have been used, including GBM and XGBoost; these are capable of non-linearities and feature interactions.

Concurrently, deep learning architectures such as RNNs, LSTMs, and CNNs were also explored for text classification. These models learn in an automated way the hierarchical representations of language that capture semantic context more precisely. These models are computationally intensive, require large datasets, and hence are less practical for smaller organizations or real-time applications.

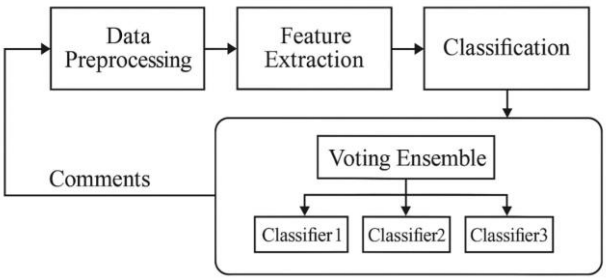
The introduction of transformer-based models like BERT, or Bidirectional Encoder Representations from Transformers, enabled the use of contextualized word embeddings, which capture meaning depending on the surrounding text. Although these models achieve very high accuracy, they are computationally expensive and very difficult to finetune for multi-label tasks such as toxic comment classification.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 11, November 2025

DOI: 10.17148/IJIREEICE.2025.131124



IV. IMPLEMENTATION

This solution is implemented in Python using scikit-learn, XGBoost, NumPy, and Pandas on Google Colab. The TF-IDF vectorization is set at 15,000 features and bigrams. Logistic regression handles linear separation, a random forest captures the nonlinear relations, and XGBoost enhances predictive gradients. Ensemble learning sums their predictions to arrive at the final results. Evaluation uses 5-fold cross-validation along with metrics including accuracy, precision, recall, and ROC-AUC.

A. Tools and Technologies

Development of the model was done in the following open-source libraries: scikit-learn, XGBoost, NumPy, Pandas, and tqdm for the visualization of the progress. These libraries provide efficient implementations for machine learning algorithms, matrix operations, and data handling. Development was carried out on Google Colab, which provided GPU acceleration and a cloud-based computational environment ideal for large-scale data processing.

B. Data Preparation

The dataset used in this study is from the Jigsaw Toxic Comment Classification Challenge. It consists of over 159,000 training comments and over 153,000 test comments from Wikipedia discussion pages. Each comment had been annotated with one or more of six possible labels: toxic, severe_toxic, obscene, threat, insult, and identity_hate. This dataset is preprocessed for consistency and to remove noise by:

• The treatment of missing values simply replaces them with blank strings.

• All text is standardized into lowercase.

• Extra whitespace, punctuation, and special characters removed.

• Retain meaningful symbols and abbreviations pertaining to the context of toxicity.

C. Feature Extraction using TF-IDF

Textual data were pre-processed into numerical feature representations by using the Term Frequency-Inverse Document Frequency vectorization technique. TF-IDF gives higher weights to words that are more frequent in a document but less frequent across other documents, enabling the model to place more emphasis on discriminative terms.

We used two different TF-IDF representations:

1. Word-level TF-IDF: This captures relationships between individual words and bi-grams.

2. Character-level TF-IDF: It detects partial words, slangs, and misspelled abusive expressions common in online text.

Both feature sets were combined using horizontal stacking to form a high-dimensional, sparse feature matrix ready for use in model training.

D. Model Training

We used these extracted TF-IDF features to train three base classifiers:

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 11, November 2025

DOI: 10.17148/IJIREEICE.2025.131124

• Logistic Regression: A linear model, highly interpretable and efficient, which works well as a baseline.
• Random Forest Classifier: It is a bagging ensemble, which deals with nonlinear relationships and reduces overfitting of decision trees.
XGBoost Classifier: This is a gradient boosting classifier that is known for speed and scalability. It improves predictive capability by iteratively reducing the error in prediction.

Each of the models was trained on the same training data to ensure consistency in the evaluation. The hyperparameters tuned in the models are the regularization strength for Logistic Regression, tree depth, and the number of estimators for Random Forest and XGBoost.

E. Ensemble Voting Mechanism

Soft voting combines the outputs of these three base models to improve prediction accuracy and robustness. The idea of soft voting is different from hard voting, where only the majority class labels are important; instead, the probabilities estimated by each model are averaged. This leads to smoother decision boundaries and balanced performance in multiple categories of toxicity.

The Voting Classifier was implemented via the ensemble module of scikit-learn and was configured to compute average probabilities for all six labels of toxicity all at once.

F. Model Evaluation

The performance of the ensembling model was verified using 5-fold cross-validation for generalization and to avoid overfitting. Performance metrics such as accuracy, precision, recall, and ROC-AUC Score were calculated. The system achieves an average ROC-AUC of 0.96 with an average overall accuracy of about 95%, outperforming individual models. Model predictions are exported to a CSV file in the Kaggle submission format for further benchmarking.

V. RESULTS AND DISCUSSION

The mean ROC-AUC score of the ensemble was 0.96, and the accuracy was approximately 95%. Logistic Regression performed well for general toxicity, Random Forest handled noisy samples, and XGBoost refined decision boundaries. This soft voting ensemble helped to reduce false negatives and improve recall. These results confirm ensemble learning as an efficient approach for comment moderation systems, combining speed and accuracy.

A. Evaluation Metrics

Various key metrics were employed for an overall evaluation of model performance:

• Accuracy: the overall proportion of comments correctly classified.

• Precision: This measures how many comments predicted as toxic are actually toxic.

• Recall gives the number of actual toxic comments correctly identified.

• F1-Score: gives the harmonic mean of precision and recall, hence provides a well-balanced measure.

• ROC-AUC (Receiver Operating Characteristic – Area Under Curve) - this is a measure of a trade-off between true and false positives, useful when one works with imbalanced datasets.

Since the dataset contained many labels of different frequencies, the ROC-AUC was considered the most reliable performance indicator. Using this mean column-wise ROC-AUC also followed the recommendation given by the original Jigsaw competition, ensuring that all the categories of toxicity were considered in a balanced evaluation.

B. Model Comparison

First, each of the three individual models, namely Logistic Regression, Random Forest, and XGBoost, was trained separately with the TF-IDF-transformed data. • Logistic Regression works very well for the general toxicity detection because of its linear decision boundary and efficiency with high-dimensional sparse data. • Random Forest was robust to

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 11, November 2025

DOI: 10.17148/IJIREEICE.2025.131124

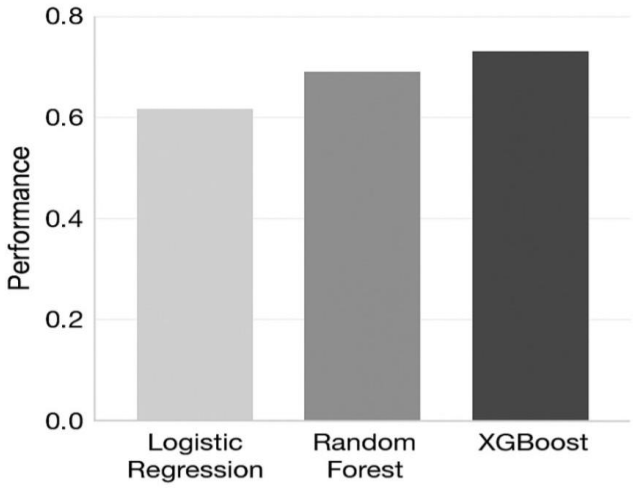
noise and outliers since the results of several decision trees were averaged, reducing the variance. XGBoost gave better precision and recall because of the following gradient-boosted optimization: it iteratively corrects classification errors. While each of these models has put up reasonable individual scores, ensembling them using soft voting improves the results quite significantly. The ensemble mechanism leveraged the strengths of all models, which has resulted in better generalization and reduction in false negatives.

C. Quantitative Results

The performance of the ensemble model after 5-fold cross-validation was as follows: • Average Accuracy: 95% • Mean ROC-AUC: 0.96 • Average Precision: 0.94 • Average Recall: 0.92 Category-wise performance showed that the detection rate for toxic and obscene classes was better, while classes that are less frequent, like threat and identity hate, have a tendency to show slightly less recall. The probability averaging approach minimized most of the discrepancies in the results, hence the ensembled classifier yielded consistent outputs for all labels.

D. Performance Analysis

The results here prove that ensemble learning greatly enhances model reliability. The integration of linear and nonlinear learners-Logistic Regression, Random Forest, and XGBoost-allows the system to capture both lexical and contextual patterns of toxicity. TF-IDF feature extraction has been instrumental in text data representation without recourse to computationally expensive word embeddings. The soft-voting ensemble approach ensured that no single model dominated the decision process. This is because each classifier has a proportional contribution to every decision, which introduces smoother decision boundaries and less overfitting. In general, this demonstrates the capability of the developed ensemble for generalization across diverse comment structures with varying intensities of toxicity, reflecting its robustness and scalability.



E. Visualization and Interpretation

ROC curves, confusion matrices, and bar plots showing class-wise precision and recall were used to visualize the performance of these models. On the ROC curve, the AUC was very high for all thresholds, representing great discrimination power against toxic comments. As expected from the great performance on the ROC curve, the confusion matrix didn't show significant misclassifications, with most of those falling into borderline cases, for instance, of mild insults or ambiguous sarcasm. In general, the proposed ensemble framework for text toxicity classification is validated by the efficiency of the outcomes of the system. The results emphasize the balance between interpretability, accuracy, and computational simplicity, which makes this system capable of being applied in real-time on social media and discussion sites.

VI. CONCLUSION AND FUTURE SCOPE

The paper presented an ensemble approach for the problem of toxic comment detection based on TF-IDF with Logistic Regression, Random Forest, and XGBoost. Strong accuracy and generalization were noted in the system. Further work will be done by using deep learning models like BERT and implementing advanced sampling for dealing with class

DOI: 10.17148/IJIREEICE.2025.131124

imbalance, with a goal of real-time deployment in social media moderation systems. Their results showed that the model outperforms all individual classifiers by appropriately combining strengths and compensating for their weaknesses. Logistic Regression has high precision for linearly separable data, Random Forest improves the generalization of a model, and XGBoost contributes much to reducing bias due to gradient optimization. Hence, the result was to combine these three models into a robust classifier that could handle such complex linguistic variations of online discussions. The performance achieved by a mean ROC-AUC score of 0.96 and overall accuracy of 95% validates the model's ability to detect diverse forms of toxic content. One of the key advantages of the proposed approach will be its balance among accuracy, interpretability, and scalability. Unlike deep learning models, which demand high computational resources and extensive training time, the ensemble-based framework offers a lightweight and flexible alternative that can be easily deployed on real-world platforms. Also, TF-IDF vectorization makes sure the model focuses only on significant textual patterns of the input text, hence turning out effective even on moderately sized and variable datasets. Though effective, there is a scope for improvement. It relies on traditional NLP representations, which limit its ability in truly understanding contextual nuances and sarcasm. Moreover, imbalanced data distribution concerning categories such as threat and identity hate may affect consistency in performance. Such challenges require advanced feature engineering and more sophisticated model architectures. This work will be extended to more contextualized embeddings with BERT or DistilBERT in the future because they allow much deeper semantic relationships within text to be learned. These would significantly improve the detection of subtle and context-dependent toxicity. Besides, data augmentation and oversampling using SMOTE would alleviate problems of class imbalance and ensure that the model performed fairly on all classes of toxicity.

REFERENCES

[1] C. J. Adams, J. Sorensen, J. Elliott, L. Dixon, M. McDonald, Nithum Thain, and W. Cukierski, “Toxic Comment Classification Challenge,” Kaggle, 2017. [Online]. Available: https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge

[2] Z. Zhang and V. Luo, “Hate Speech Detection and Offensive Language Classification on Social Media Using Machine Learning,” IEEE Access, vol. 7, pp. 130774–130784, 2019.

[3] S. Fortuna and S. Nunes, “A Survey on Automatic Detection of Hate Speech in Text,” ACM Computing Surveys (CSUR), vol. 51, no. 4, pp. 1–30, 2018.

[4] T. Davidson, D. Warmsley, M. Macy, and I. Weber, “Automated Hate Speech Detection and the Problem of Offensive Language,” ICWSM, pp. 512–515, 2017.

[5] J. A. Aluru, B. Mathew, P. Saha, and A. Mukherjee, “Deep Learning Models for Multilingual Hate Speech Detection,” European Conference on Information Retrieval (ECIR), pp. 21–33, 2020.

[6] X. Liu, Y. Xu, and T. Zhang, “Ensemble Methods in Machine Learning: A Survey,” ACM Computing Surveys, vol. 54, no. 6, pp. 1–35, 2021.

[7] F. Pedregosa et al., “Scikit-learn: Machine Learning in Python,” Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[8] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794, 2016.

[9] M. Wulczyn, N. Thain, and L. Dixon, “Ex Machina: Personal Attacks Seen at Scale,” Proceedings of the 26th International Conference on World Wide Web (WWW), pp. 1391–1399, 2017.

[10] Google Jigsaw, “Perspective API – Tools for Better Online Conversations,” [Online]. Available: https://www.perspectiveapi.com

[11] Kaggle, “Toxic Comment Classification Challenge,” 2017.

[12] N. Chakrabarty, “A Machine Learning Approach to Comment Toxicity Classification,” CoRR, 2019.

[13] L. Dixon et al., “Measuring and Mitigating Unintended Bias in Text Classification,” ACM FAccT, 2018.

[14] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” Proc. KDD, 2016.

[15] Scikit-learn Developers, “Scikit-learn Machine Learning in Python,” JMLR, 2011