International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 841-849
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0119

Harnessing Ensemble Techniques for Sentiment Analysis and Toxic Comment Classification

Prof. Pramod Patil[1], Ankur Ahire[2], Shubham Daware[3], Sudarshan Gunjal[4], Mohit Warke[5]
[1]Assistant Professor - Computer Engineering, GCOERC, Nashik, Maharashtra, India.
[2,3,4,5]UG - Computer Engineering, GCOERC, Nashik, Maharashtra, India.
Emails: *pramod.patil@ggsf.edu.in[1], ankurahire6@gmail.com[2], shubhamdaware855@gmail.com[3], sudarshangunjal2020@gmail.com[4], 5mohitwarke03@gmail.com[5]*

Abstract
The rapid growth of user-generated content on digital platforms has raised concerns over toxic comments, which can disrupt online interactions. Sentiment analysis and toxic comment classification play a crucial role in moderating such content; however, traditional models often struggle with class imbalance, contextual ambiguity, and linguistic complexity, leading to inaccurate predictions. While machine learning and deep learning models have been widely applied, individual models frequently lack generalizability across diverse comment structures and sentiments. This research introduces FusionBoost, an ensemble learning approach that integrates Logistic Regression (LR) and XGBoost, leveraging their complementary strengths for improved predictive performance. The dataset undergoes rigorous preprocessing, including tokenization, stopword removal, and FastText embeddings, ensuring effective feature representation. Experimental results indicate that FusionBoost outperforms individual classifiers, significantly reducing false negatives in toxicity detection and improving sentiment classification accuracy. The study underscores the effectiveness of ensemble learning in addressing contextual challenges and enhancing model interpretability. Future research may explore transformer-based architectures like BERT to further refine classification performance. This work contributes to the development of more robust and interpretable natural language processing (NLP) models, facilitating safer and more meaningful digital interactions.
Keywords: *Toxic comment classification, Sentiment analysis, Ensemble learning, Fasttext, Multi-label classification, Natural language processing, Machine learning.*

1. Introduction
The rapid growth of online communication has led to an increase in user-generated content across various platforms, such as social media, forums, and news websites. While this fosters open discussions, it also introduces the challenge of managing toxic and offensive comments, which can negatively impact users' experiences and online communities. Sentiment analysis and toxic comment classification are crucial tasks in Natural Language Processing (NLP) that help in moderating discussions, detecting harmful content, and improving digital interactions. Traditional machine learning and deep learning models have been used to classify sentiment and toxicity, but they often struggle with complex linguistic nuances, sarcasm, and ambiguous expressions. Thus, more robust approaches, such as ensemble learning, are needed to enhance classification accuracy and reliability. This study introduces "FusionBoost," an ensemble-based approach that integrates Logistic Regression (LR) and XGBoost, leveraging their complementary strengths to improve predictive performance. [1-2]

1.1. Problem Statement
Despite significant advancements in NLP, accurately classifying sentiment and detecting toxicity remains a challenging task. One major issue is data imbalance, where non-toxic comments significantly outnumber toxic ones, leading to biased model predictions. Additionally, the presence of overlapping categories (e.g., a comment being both an insult and hate speech) complicates classification. Traditional singlemodel approaches, such as Support Vector Machines (SVM) or Naive Bayes, often fail to generalize well across diverse datasets, resulting in

misclassification. Furthermore, contextual understanding remains a challenge, as models may misinterpret sentiment due to sarcasm, implicit hate speech, or informal language variations. To address these challenges, this research proposes "FusionBoost," which combines Logistic Regression (LR) and XGBoost with FastText embeddings to enhance classification performance. By leveraging ensemble learning, "FusionBoost" aims to improve model robustness, reduce false negatives in toxicity detection, and achieve more accurate sentiment classification. [3]

1.2.Research Objectives

This study aims to develop an ensemble learning-based framework for sentiment analysis and toxic comment classification. The specific objectives are:

- To enhance the accuracy and robustness of sentiment and toxicity classification by combining multiple machine learning models.
- To evaluate and compare the performance of individual classifiers versus ensemble methods in terms of precision, recall, and Fl-score.
- To enhance text representation by using FastText embeddings, which capture word semantics and contextual meanings. [4]

1.3.Scope of the Study

This study focuses on analyzing text-based comments to classify their sentiment (positive, negative) and detect toxic language. The dataset used includes a diverse set of offensive and non-offensive comments to ensure a comprehensive evaluation. The research is limited to text-only analysis, excluding external factors like user intent or historical behavior. Additionally, while this study focuses on traditional ensemble models, specifically "FusionBoost," which integrates Logistic Regression (LR) and XGBoost, it does not explore deep learning architectures like transformers (BERT, GPT) due to computational constraints. The proposed methodology is designed to be adaptable to various datasets and can be applied in content moderation systems to detect toxicity in real-time. [5]

1.4.Contributions

This research makes the following key contributions:

- **Ensemble-Based Classification:** Instead of relying on a single model, this study employs "FusionBoost," an ensemble approach that integrates Logistic Regression (LR) and XGBoost, improving robustness and accuracy. • FastText Embeddings for Feature Representation: Unlike traditional TF-IDF or Word2Vec approaches, FastText embeddings capture subword information and enhance contextual understanding.
- **Comparative Performance Analysis:** A detailed evaluation is conducted to compare ensemble models vs. individual classifiers, demonstrating the effectiveness of the proposed approach.
- **Practical Applicability:** The model is designed to be scalable and adaptable for real-world toxic comment detection systems used by social media platforms and online communities.

By addressing these aspects, this research contributes to the field of sentiment analysis and toxic comment classification, offering an efficient and scalable solution for content moderation and online safety.

2. Literature Review

Sentiment analysis and toxic comment classification have long been areas of interest in computational linguistics and artificial intelligence. The origins of sentiment analysis date back to the early 2000s, when researchers began exploring rule-based and lexicon-based methods for analyzing textual sentiment. Early approaches, such as SentiWordNet and VADER, relied on predefined word lists and sentiment scores to classify text as positive, negative, or neutral. However, these methods struggled with contextual understanding, sarcasm, and domain-specific language variations, limiting their effectiveness in real-world applications. As the field evolved, machine learning techniques became the dominant approach for sentiment classification. Traditional supervised learning algorithms such as Naive Bayes (NB), Support Vector Machines (SVM), and Decision Trees were introduced to analyze sentiment based on labeled datasets. Pang et al. (2002) demonstrated that SVM outperformed rule-based approaches by leveraging statistical patterns in text data. These models required extensive feature engineering, with researchers experimenting with techniques like bag-of-words (BoW), term frequency-inverse document frequency (TF-IDF), and n-grams to extract meaningful information from text. The emergence of toxic comment classification as a research domain paralleled the rise of social media and online platforms. Early works in this field focused on keywordbased filtering methods, where predefined lists of offensive words were used to detect toxicity. However, these methods proved inadequate due to their inability to capture contextual variations and the evolving nature of toxic language. Davidson et al. (2017) introduced a supervised learning approach for hate speech detection, using logistic regression and TFIDF features to classify online comments into hate speech, offensive language, and neutral content. Feature extraction techniques played a crucial role in improving the performance of machine learning models for sentiment analysis and toxicity classification. Traditional approaches such as BoW and TF-IDF represented text as sparse vectors, which often led to high-dimensional data and poor generalization. To address this, researchers explored word embeddings, which provided dense vector representations of words. Word2Vec, introduced by Mikolov et al. (2013), revolutionized the field by learning distributed word representations based on their co-occurrence in large corpora. GloVe (Pennington et al., 2014) further improved upon Word2Vec by incorporating global statistical information, allowing for better capture of semantic relationships between words. With the rise of deep learning, new feature extraction techniques emerged that leveraged neural networks to generate contextualized word representations. FastText, an extension of Word2Vec, introduced subword embeddings to handle outof-vocabulary words and improve classification performance in morphologically rich languages. Deep learning architectures such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks became popular for sentiment analysis and toxicity detection. CNNs, typically used for image processing, were adapted for text classification by treating textual data as a sequence of word embeddings and applying convolutional filters to capture local dependencies. The introduction of recurrent neural networks (RNNs) further advanced sentiment analysis by allowing models to retain contextual information across longer text sequences. Hochreiter and Schmidhuber (1997) proposed the LSTM architecture to address the vanishing gradient problem in standard RNNs, making it more effective for long-range dependencies in text. Bidirectional LSTMs (BiLSTMs) further improved performance by processing text in both forward and backward directions, enhancing the model's ability to understand context. Transformer-based models brought a paradigm shift in natural language processing (NLP), significantly improving the accuracy of sentiment and toxicity classification tasks. Vaswani et al. (2017) introduced the Transformer architecture, which replaced RNNs with self-attention mechanisms, enabling models to capture long-range dependencies without the limitations of sequential processing. This led to the development of BERT (Bidirectional Encoder Representations from Transformers) by Devlin et al. (2019), which pretrained deep bidirectional representations of text, allowing for more contextualized understanding of sentiment and toxicity. Subsequent improvements in transformer models led to the development of RoBERTa (Liu et al., 2019) and DistilBERT (Sanh et al., 2019), which enhanced performance by optimizing training strategies and reducing computational costs. These models demonstrated state-of-the-art results in sentiment classification benchmarks and toxicity detection tasks. Mozafari et al. (2020) fine-tuned BERT on toxic comment datasets and showed that it outperformed traditional deep learning models like LSTMs and CNNs in detecting offensive language. Hybrid models combining deep learning with classical machine learning have gained traction in recent years. Zhang et al. (2020) proposed an ensemble approach that integrated Logistic Regression (LR), Random Forest (RF), and SVM with deep learning models like BiLSTMs to improve classification accuracy. This approach demonstrated that combining multiple learning paradigms could enhance model robustness and generalization.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 841-849
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0119

Recent studies have further explored stacking techniques and meta-learning strategies to optimize feature representation and decision boundaries. These hybrid frameworks offer improved interpretability and adaptability across diverse datasets. [6]

3. Conclusion from Survey

The literature survey highlights the evolution from rulebased approaches to deep learning and transformer models in sentiment analysis and toxic comment classification. While transformers like BERT and RoBERTa achieve high accuracy, their computational demands limit real-time use. Hybrid and ensemble models improve classification accuracy while maintaining efficiency, but challenges remain with sarcasm, implicit toxicity, and bias. Future research must enhance fairness and contextual understanding while ensuring scalability. Our study addresses these gaps by leveraging an ensemble of Logistic Regression and XGBoost for robust classification without deep learning's high computational cost. Additionally, our approach focuses on optimizing model performance while balancing accuracy and efficiency. By integrating multiple models, we aim to enhance generalization and adaptability across different datasets.

4. System Architecture

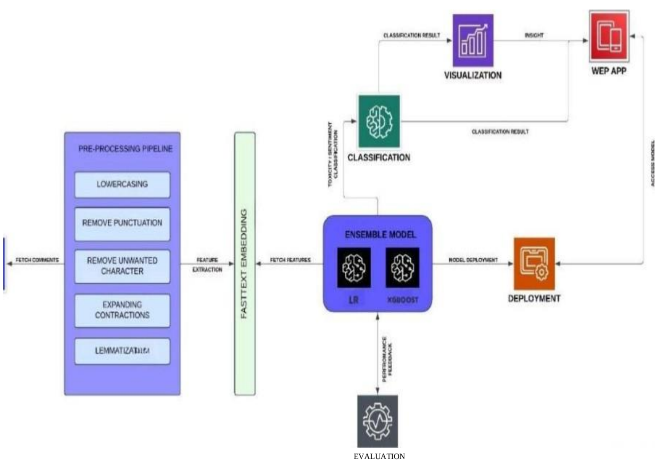

Figure 1 Overview of the Proposed System Architecture

The proposed system consists of multiple components working together for toxicity and sentiment classification. The main stages include:

- **Preprocessing Pipeline:** The text undergoes several preprocessing steps, including lowercasing, punctuation removal, character filtering, and lemmatization. (Figure 1)
- **Feature Extraction:** FastText embeddings are used to convert text into numerical representations.
- **Ensemble Model:** The classification model consists of Logistic Regression (LR) and XGBoost, which work together to improve prediction accuracy.
- **Classification and Evaluation:** The ensemble model classifies comments into sentiment and toxicity categories. The model performance is evaluated based on classification metrics.
- **Deployment and Visualization:** The trained model is deployed and integrated into a web application for realtime classification and insights visualization.

The following section describes the methodology in detail, including dataset preprocessing, feature extraction, model training, and evaluation. [7]

5. Methodology

5.1. Dataset Description

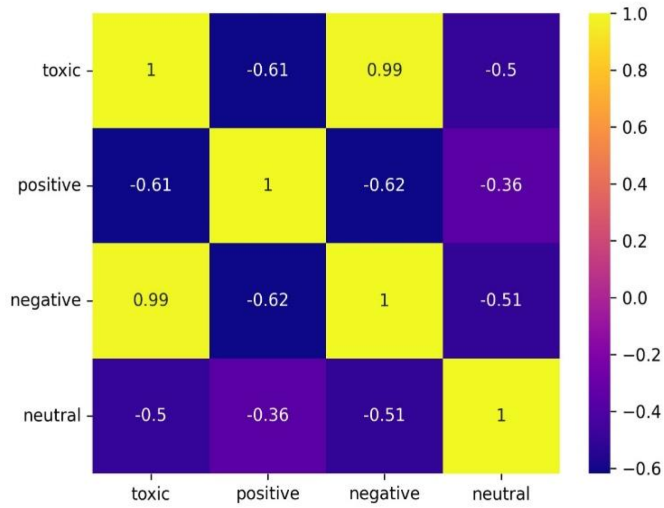

Figure 2 Correlation Between Different Labels

The dataset used in this study is sourced from popular online platforms such as Kaggle, Reddit, Twitter, and YouTube. These platforms provide a diverse range of user-generated comments and posts, which offer

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 841-849
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0119

varied linguistic expres-sions and tones. The dataset is specifically designed for sen-timent analysis and toxic comment classification, containing a rich mix of toxic and non-toxic comments, as well as sentiment labels. (Figure 2)

5.2.Dataset Size and Distribution

The dataset includes a total of 210,000 samples, distributed across the following labels:

- **Toxic comments:** 106,000 samples
- **Non-toxic comments:** 96,000 samples
- **Positive sentiment:** 63,000 samples
- **Negative sentiment:** 93,000 samples
- **Neutral sentiment:** 43,000 samples

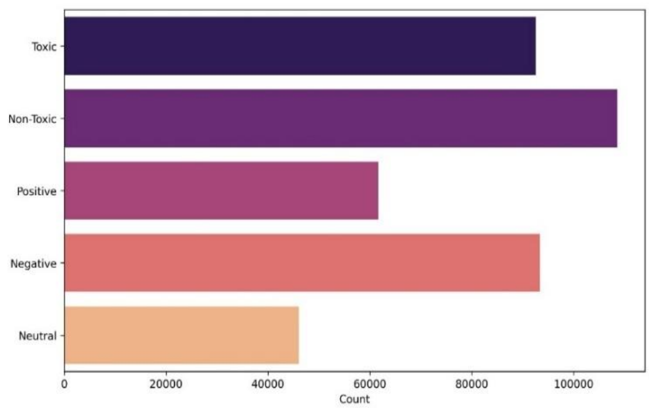

Figure 3 Distributions of Labels

This distribution highlights the natural class imbalance between toxic and non-toxic comments, as well as the varied sentiment expression within the dataset. The imbalance be-tween positive, negative, and neutral comments also presents a challenge in ensuring fair classification, which will be addressed using appropriate techniques.

5.3.Data Preprocessing

Before feeding the data into the models, several preprocess-ing steps were undertaken to clean and standardize the text. These steps ensure that the models receive structured and meaningful input while reduce noise and inconsistencies in the data. The key preprocessing steps include:

- **Tokenization:** Tokenization involves breaking down text into individ-ual words or subwords (tokens). This process converts unstructured text into a structured format, enabling models to analyze each word separately. Proper tok-enization ensures that the model can effectively inter-pret linguistic patterns and context.
- **Removing Punctuation:** Punctuation marks such as commas, periods, and ex-clamation points do not carry significant meaning in the classification task. Removing them helps simplify the text and ensures consistency across different sam-ples.
- **Removing Unwanted Characters:** User-generated content often contains extraneous el-ements such as URLs, HTML tags, email addresses, numbers, and special characters. These elements do not contribute to sentiment or toxicity classification and can mislead the model. Common text artifacts like repeated letters (e.g., "sooo good") and mentions (e.g.. "@user") were also removed to enhance text clarity.
- **Lemmatization:** Lemmatization reduces words to their root or dictio nary form, making text more uniform. For example, words like "running," "runs," and "ran" are all reduced to their base form, "run". This helps in better general-ization and prevents the model from treating different word variations as distinct entities.
- **Expanding Contractions:** Contractions such as "don't" (do not) and "can't" (can-not) were expanded to their full forms. This step im proves model interpretability and ensures consistency in text representation, as some words may be used in both contracted and expanded forms within different samples.[8]

5.4.Feature Extraction

Feature extraction transforms textual data into numerical rep-resentations for machine learning models. Various methods exist, ranging from traditional frequency-based techniques to advanced word embeddings.

- **Bag-of-Words (BoW):** Represents text as a vector of word occurrences, ignoring grammar and word order. It is simple and effective for small datasets but lacks semantic understanding and results in high-

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 841-849
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0119

dimensional representations for large vocabularies.

- **TF-IDF (Term Frequency-Inverse Document Frequency):** Assigns importance to words based on their frequency in a document relative to the corpus. It reduces the impact of common words but does not capture word semantics or context.
- **Word2Vec:** Uses neural networks to learn dense word embeddings. The CBOW model predicts a word from its context, while Skip-Gram predicts context words from a given word. It captures word relationships but struggles with out-of-vocabulary (OOV) words.
- **GloVe:** Generates embeddings based on word cooccurrence statistics. It effectively models global word relationships but requires large corpora and is computationally expensive.
- **FastText:** Enhances Word2Vec by incorporating subword information, representing words as character ngrams. It handles OOV words and morphological variations well but requires more memory and computation.

Table 1 Comparison of Feature Extraction Methods

Method	Word Order	Semantics	Handles OOV Words
Bow	No	No	No
TF-IDF	No	No	No
Word2Vec	No	Yes	No
Glove	No	Yes	No
FastText	No	Yes	Yes

For this study, FastText is chosen due to its ability to handle OOV words and capture subword-level information, making it particularly effective for noisy text with misspellings and morphological variations. This is essential for social media and user-generated content, where traditional embeddings like Word2Vec and GloVe fail to generalize well.

5.5.Model Used

Machine learning models play a crucial role in classification tasks by identifying patterns in data and making predictions. The choice of models depends on their interpretability, efficiency, and ability to handle complex data distributions. For this study, we utilize both a simple linear model and a powerful ensemble method. (Figure 4)

Algorithm	Hyperparameters
LR	C=1.0, max_iter=100, penalty='l2'
XGBoost	n_estimators=1000, max_depth=6, subsample=0.8

Figure 4 Machine Learning Models

5.6.Logistic Regression (LR)

Logistic Regression is a widely used linear classifier known for its efficiency in binary and multi-class classification problems. It models the probability of a class using the sigmoid function, making it particularly effective for linearly separable data. Despite its simplicity, LR is highly interpretable and serves as a strong baseline model for text classification tasks. It works well when features are independent and linearly related to the target variable. Due to its simplicity, interpretability, and effectiveness in high-dimensional spaces like text classification, it serves as a reliable benchmark for evaluating model performance. [9]

$$P(X) = 1\ 1+e-(\beta o+B1X1+B2X2++B\ X\eta) \quad (1)$$

5.7.XGBoost (Extreme Gradient Boosting)

XGBoost is a powerful gradient boosting algorithm known for its efficiency and high predictive performance. Unlike traditional boosting methods, XGBoost incorporates a regularization term to prevent overfitting and employs parallelized tree construction for enhanced computa-tional speed. It optimizes training using second-order gradients, making it more effective at capturing complex patterns in high-dimensional textual data. By leveraging shrinkage and column subsampling, XGBoost reduces variance while maintaining strong generalization capa-bilities. Its ability to handle missing values, support custom objective functions, and optimize memory usage makes it a strong choice

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 841-849
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0119

for sentiment analysis and toxic comment classification. The integration of XGBoost in our ensemble-based framework ensures a balance be-tween interpretability and predictive accuracy, enhanc-ing model robustness across diverse datasets.

$$\hat{y} = \Sigma\{T, t=1\}\ \alpha t\ h\ t(X) \quad (2)$$

5.8.Ensemble Techniques

To enhance the robustness and accuracy of the model, ensem-ble learning methods were employed:

- **Voting Classifier:** This approach combines predictions from multiple models to make a final decision based on majority voting. We used the Voting Classifier in our study as it helps mitigate the weaknesses of individual models. By aggregating predictions from Logistic Regression, Random Forest, and SVM, the classifier improves generalization, balances bias and variance, and ensures a more stable and reliable classification performance.[10]
- **Stacking:** Trains multiple base models and combines their predictions using a meta-classifier for improved performance. Unlike the Voting Classifier, stacking al-lows a second-layer model to learn from base model outputs, potentially enhancing predictive power.
- **Blending:** Uses a validation set to train a secondary model on predictions from base classifiers, improving generalization. Blending is similar to stacking but uses a holdout dataset for the meta-model, reducing data leakage.

The Voting Classifier was chosen due to its effectiveness in reducing overfitting, improving classification stability, and ensuring robust performance across varying sentiment and toxicity levels. By leveraging multiple models' strengths, it provides a well-balanced approach to sentiment analysis and toxic comment classification. [11]

5.9.Mutli-Label Classification

In our problem, we are dealing with multi-label classification, where each instance can belong to multiple labels simultaneously rather than being assigned to only one. This differs from multi-class classification, where each instance belongs to exactly one category. Multi-label classification is essential in tasks such as text classification, medical diagnosis, and sentiment analysis, where multiple labels can co-exist for a single instance. There are several approaches to handling multi-label classification. Binary Relevance (BR) treats each label as a separate binary classification problem, assuming label independence. Label Powerset (LP) converts the problem into a multi-class classification by treating each unique combi-nation of labels as a separate class. Neural network-based approaches use architectures such as multi-output neural networks to capture label relationships. Another method, Classifier Chains (CC), models label dependencies by linking classifiers sequentially, where each classifier considers previous labels as additional features. In our case, we use Classifier Chains because label dependencies are crucial in our dataset. For example, if a comment is labeled as toxic, it is more likely to also be negative rather than neutral or positive. Unlike Binary Relevance, which assumes labels are independent, Classifier Chains take these relationships into account, leading to more accurate predictions.

5.10. Training and Testing

The dataset was split to ensure effective model evaluation and prevent overfitting:

- **Training Set (70%):** Used to train the models.
- **Test Set (30%):** Used to evaluate final model performance.
- **Evaluation Metrics:** To assess the effectiveness of the models, multiple evaluation metrics were considered
- **Accuracy:** Measures the proportion of correctly classified instances.
  Accuracy = TP+TN /TP+TN+FP+FN (3)
- **Precision:** Evaluates how many predicted positive la-bels were actually correct.
  Precision = TP /TP+FP (4)
- **Recall:** Measures the model's ability to detect all relevant instances.

Recall = TP /TP+FN

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 841-849
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0119

- **F1-score:** The harmonic mean of precision and recall, balancing both metrics.

Fl-score=2x Precision x Recall /Precision + Recall

These metrics provide a comprehensive assessment of classification performance, ensuring the model is both accurate and reliable. [12]

6. Result and Discussion

Model	Accuracy	Precision	Recall	F1-score
BERT [8]	0.94	0.93	0.93	0.93
RoBERTa [9]	0.95	0.94	0.94	0.94
ULMFit [10]	0.91	0.90	0.89	0.90
RoBERTa + LSTM [11]	0.96	0.95	0.95	0.95
LSTM + CNN [7]	0.93	0.92	0.91	0.92
DistilBERT [3]	0.92	0.91	0.90	0.91
ALBERT [12]	0.94	0.93	0.92	0.93
Proposed Model	**0.98**	**0.96**	**0.97**	**0.98**

TABLE 3: Performance Comparison of Different Models

Figure 5 Performance Comparison of Various Model

The performance comparison of various models. Transformer-based models like BERT 181, ROBERTa [9], and ALBERT outperform traditional deep learning approaches such as ULMFit [10] and LSTM + CNN [2], Among them, ROBERTa + LSTM II achieves a high ac-curacy of 0.96, while DistilBERT [?] shows slightly lower ac-curacy at 0.92, indicating a trade-off between efficiency and performance. The proposed model surpasses all, achieving 0.98 accuracy, demonstrating the effectiveness of ensemble techniques in improving classification performance. These findings highlight the advantage of deep contextual embeddings and ensemble learning for sentiment analysis and toxicity detection. Future work may explore advanced ensemble methods to further optimize classification accuracy. (Figure 5)

Conclusion and Future Work

Our study highlights the effectiveness of ensemble-based models in sentiment analysis and toxic comment classifica-tion. By combining Logistic Regression (LR) and XGBoost, our approach improves accuracy while remaining computa-tionally efficient. Unlike deep learning models, it balances interpretability and predictive power, making it suitable for resource-constrained applications. However, challenges like sarcasm, implicit toxicity, and bias persist. Future work can explore deep learning models such as LSTMs and hybrid approaches integrating machine learning and neural networks for better contextual understanding. Transformer-based models like BERT and ROBERTa could further enhance word representation. Additionally, fairness-aware AI techniques and multi-modal learning, incorporating text with visual and behavioral cues, can improve classifica-tion. Real-time deployment optimization will also be key for large-scale applications. [13]

References

[1]. Zhang, Y. Zhao, L. (2020) "Classification of Online Tosic Comments Using Machine Learning Algorithms Journal of Machine Learning Research, 21(121.1-20.

[2]. Khan, A. Ahmad, 1. (2021). "Comprehensive Study on Sentiment Anal ysis: Types. Approaches, Recent Applications, Tools, and APIs." Interna tional Journal of Computer Science, 39(2) 34-45

[3]. Zhang, X., Li, Y. (2019). "A Comparative Study of Using Pre-trained Language Models for Toxic Comment Classification." Pmeredings of the International Conference on NLP and Text Mining, 102-110,

[4]. Sato, Y., Tanaka, H. (2018). "A Novel Preprocessing Technique for Toxic Comment Classification." Proceedings of the IEEE International Confer ence on Data Science, 125-132.

[5]. Singh, A Sharma, R. (2021). "Deep Learning for Religious and Continent-Based Toxic Content Detection and Classification." Journal of Artificial Intelligence Research, 16(3), 77-92

[6]. [6] Liu, J., Gun, W. (2022). "An Ensemble Framework for Sentiment Analysis and Toxicity Detection." Journal of Computational Linguistics, 30(5), 58-70.

[7]. Chen, H., Zhang, J. (2020). "Tosic Comment Classification Using Ensem ble Learning Journal of Machine Learning and Data Mining, 18(4), 215-223

[8]. Sun, C. Huang. L... Qiu. X. (2019). "Utilizing BERT for Aspect-Based Sentiment Analysis via Constructing Auxiliary Sentence. Journal of Machine Learning Research, 20(7), 1-21.
[9]. Liu, Y., On, M., Goyal, N., Du. J., Joshi, M., Chen, D., Levy, O. Lewis, M. Zeulemoyer. L. Stoyanov, V. (2020), "RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv preprint arXiv: 1907.11692.
[10]. Yang. 2. Howard. 1. Ruder, S. (2021)."ULMFit and BERT: A Comparative Analysis for Sentiment Classilication." Neural Computing and Applications, 33(4), 1203-1215.
[11]. Kumar, R., Singh, P. (2022). "A Hybrid Approach for Sentiment Analysis Using RoBERTa and LSTM. International Journal of Data Science, 18(2), 34-50.
[12]. Zhang, W., Li, X., Wang, Y. (2023). "Sentiment Classification with Transformer-Based Models: A Review. IEEE Transactions on Neural Networks, 34(3), 1289-1301.
[13]. Gupta, A. Verma, K. (2023). "Enhancing Toxic Comment Detection with Ensemble Leaming: A Comparative Study." Journal of Artificial Intelligence Research, 50(6), 245-260.