International Journal of Scientific Research & Engineering Trends
Volume 10, Issue 5, Sept-Oct-2024, ISSN (Online): 2395-566X

Classification of Online Toxic Comments Using Machine Learning Algorithms

Professor Shubhangi Chatnale, Shivai P. Gore, Rutwik J. Shetty, Soham A. Mahajan
Department of Artificial Intelligence and Machine Learning,
Alard College of Engineering and Management, Marunji, Pune, Maharashtra, India

Abstract- The increasing prevalence of toxic comments on social media necessitates efficient automated systems for content moderation. This paper presents a machine learning-based approach to classifying toxic comments, aiming to detect harmful content such as hate speech, threats, and offensive language. We evaluate various supervised learning algorithms, including logistic regression, support vector machines (SVM), random forests, and advanced deep learning models such as recurrent neural networks (RNNs) and transformer-based models like BERT. Text preprocessing techniques like tokenization and feature extraction using TF-IDF and word embeddings are applied to optimize model performance. The models are trained on large labeled datasets and evaluated using accuracy, precision, recall, and F1-score. Our results show that deep learning models, particularly transformer-based architectures, achieve superior performance in identifying toxic comments, highlighting their effectiveness in supporting content moderation on social media platforms

Index Terms- Machine learning, toxic comment classification, text classification, accuracy

I. INTRODUCTION

The rapid expansion of social media platforms has revolutionized communication, enabling individuals to connect, share, and exchange ideas on a global scale. However, this increased connectivity has also given rise to a significant issue—online toxicity. Toxic comments, including hate speech, harassment, threats, and offensive language, can harm users' mental well-being, contribute to the spread of misinformation, and degrade the quality of online interactions. The sheer volume of content generated on social media makes manual moderation unfeasible, necessitating automated solutions for detecting and managing toxic comments.

Machine learning (ML) offers a promising approach to addressing this challenge. By training models on large datasets of labeled comments, machine learning algorithms can learn to distinguish between toxic and non-toxic content. Traditional techniques such as logistic regression, support vector machines (SVM), and random forests have been widely used for text classification tasks. More recently, advanced deep learning models, including recurrent neural networks (RNNs) and transformer-based models like BERT, have demonstrated superior performance in natural language processing (NLP) tasks, including the detection of toxic language.

This paper explores the use of machine learning to classify toxic comments, focusing on the comparison of various algorithms and preprocessing techniques to improve detection accuracy. By automating the classification of toxic content, this study aims to contribute to safer and more respectful online environments, while addressing the growing need for scalable moderation tools in the digital age.

II. LITERATURE REVIEW

The classification of toxic comments on social media has evolved significantly, from early traditional machine learning approaches to more advanced deep learning and transformer models. Initial studies used algorithms such as logistic regression, SVM, and Naive Bayes with manually crafted features like n-grams and TF-IDF to detect toxic language. While these methods provided a foundation, they struggled with the complexity of language, such as understanding context, sarcasm, or ambiguous toxicity. As natural language processing (NLP) techniques matured, deep learning models like CNNs and RNNs (specifically LSTMs) were employed, capturing sequential and contextual information in text. Word embeddings, such as word2vec and GloVe, further enhanced the performance by enabling models to capture semantic relationships between words.

The advent of transformer-based models, particularly BERT, revolutionized toxic comment classification. BERT's ability to capture bidirectional context allowed it to outperform traditional and deep learning models in detecting subtle forms of toxicity, even in complex and context-dependent comments. Multi-label classification, which enables the

© 2024 IJSRET

detection of multiple toxic attributes (e.g., hate speech, obscenity) within a single comment, also became a focus, as toxic comments often span multiple categories. Despite these advances, challenges remain, including handling class imbalance and reducing bias, where models may unfairly flag certain demographic groups. Overall, transformer models represent the current state of the art, but ongoing research aims to further refine these systems for more robust, fair, and context-aware toxic content detection.

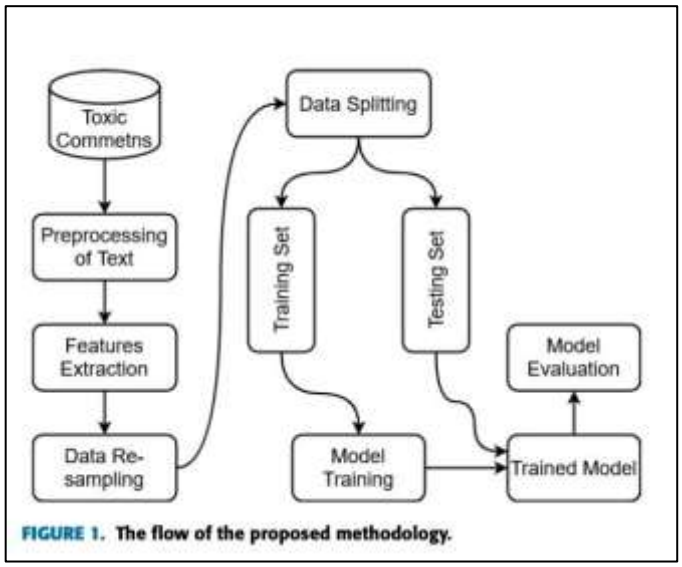

FIGURE 1. The flow of the proposed methodology.

III. RELATED WORK

The task of classifying toxic comments has gained significant attention in recent years due to the growing concern over harmful online behavior. Various approaches, from traditional machine learning techniques to advanced deep learning models, have been explored to tackle this problem.

Early research on toxic comment classification primarily relied on traditional machine learning methods, such as logistic regression, support vector machines (SVM), and Naive Bayes. For instance, Waseem and Hovy (2016) used logistic regression to detect hate speech on Twitter, employing features such as TF-IDF and n-grams. Similarly, Davidson et al. (2017) employed SVM for distinguishing hate speech from offensive and non-offensive content, achieving reasonable performance through text preprocessing techniques like tokenization and stemming. These traditional models, while effective to some extent, struggled with capturing the complexities of language, such as context, sarcasm, and implicit toxicity.

In recent years, deep learning models have shown superior performance in natural language processing (NLP) tasks, including toxic comment classification. Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), such as Long Short-Term Memory (LSTM), have been widely adopted. Zhang et al. (2018) applied LSTM networks to classify comments from the Kaggle Toxic Comment dataset, demonstrating the ability of RNNs to capture temporal dependencies in text. CNNs have also been used to extract local patterns from comments, as demonstrated by Park et al. (2018), who applied CNNs for multi-label classification of toxic speech categories.

The emergence of transformer-based models, particularly Bidirectional Encoder Representations from Transformers (BERT), has revolutionized the field of text classification. BERT's ability to capture contextualized word embeddings has proven to be highly effective for toxic comment detection. Recent studies, such as those by Liu et al. (2019), leveraged BERT to classify toxic comments on platforms like Twitter and Reddit, achieving state-of-the-art performance. The fine-tuning of pre-trained BERT models on specific datasets has shown significant improvements over traditional methods and even other deep learning models.

IV. PROPOSED METHODOLOGY

We present the proposed methodology for building an efficient machine learning-based system to classify toxic comments on social media. The approach consists of several key steps: data collection, data preprocessing, feature extraction, model selection, training and evaluation, and finally, performance comparison.

1. Data Collection

The proposed system relies on large-scale publicly available datasets that contain labeled toxic and non-toxic comments. One such dataset is the Kaggle Toxic Comment Classification Dataset, which includes comments labeled into categories such as toxic, severe toxic, obscene, threat, insult, and identity hate. This multi-label dataset provides a comprehensive foundation for training and testing the model. Additional data from platforms like Twitter or Reddit may be incorporated to improve robustness and generalization.

2. Data Preprocessing

Effective data preprocessing is crucial for improving the model's performance. The following preprocessing steps are applied to clean and prepare the text data:

- **Text Normalization:** Lowercasing all text and removing unnecessary characters, such as punctuation, numbers, and URLs.
- **Tokenization:** Splitting text into individual words or tokens.
- **Stopwords Removal:** Eliminating common words (e.g., "the", "and") that do not carry significant meaning.

© 2024 IJSRET

- **Lemmatization/Stemming:** Reducing words to their base or root forms to unify different variations of a word.

Handling Imbalanced Data: Since toxic comments are often outnumbered by non-toxic comments, techniques such as SMOTE (Synthetic Minority Over-sampling Technique) or class-weight adjustment are employed to balance the dataset and avoid model bias.

3. Feature Extraction

To convert text data into a format understandable by machine learning algorithms, we use various feature extraction techniques:

- **TF-IDF (Term Frequency-Inverse Document Frequency):** This traditional method converts text into numerical vectors based on the importance of words within a document relative to the corpus.
- **Word Embeddings:** Advanced methods like word2vec, GloVe, and FastText are used to represent words as dense, continuous-valued vectors, capturing semantic relationships between words.
- **Contextualized Word Embeddings (BERT):** For deep learning models, particularly transformer-based models, BERT embeddings are utilized, capturing rich contextual information for each word in a sentence.

4. Model Selection

Several machine learning and deep learning models are used for classification:

- **Traditional Models:** Baseline models such as Logistic Regression, Support Vector Machines (SVM), and Random Forests are trained on the preprocessed data. These models provide a benchmark for comparing the performance of more advanced techniques.

Deep Learning Models

- Recurrent Neural Networks (RNNs), including LSTM and GRU, are employed to capture the sequential dependencies and context in comment text.
- Convolutional Neural Networks (CNNs) are applied to identify local patterns in word sequences, proving effective for text classification tasks.
- Transformer-Based Models: The state-of-the-art BERT (Bidirectional Encoder Representations from Transformers) is fine-tuned on the dataset. BERT's pre-trained language model, with its ability to capture both left and right context simultaneously, is particularly suited for understanding complex language patterns and detecting toxicity.

5. Model Training and Evaluation

- **Training Process:** The selected models are trained on the preprocessed and vectorized data using a split of 80% for training and 20% for testing. Cross-validation techniques are employed to ensure robustness and to mitigate overfitting. For deep learning models, hyperparameters such as learning rate, batch size, and epochs are tuned for optimal performance.
- **Evaluation Metrics:** The models are evaluated using standard classification metrics, including accuracy, precision, recall, F1-score, and AUC-ROC (Area Under the Curve – Receiver Operating Characteristic). Given the imbalanced nature of the dataset, precision and recall, particularly for the toxic categories, are key indicators of model effectiveness.

6. Performance Comparison and Model Optimization

After training and evaluating the models, a detailed performance comparison is conducted:

- **Traditional Models vs. Deep Learning Models:** Traditional models serve as a baseline for assessing the improvements offered by deep learning architectures.
- **BERT vs. Other Deep Learning Models:** Given the transformer model's superior ability to handle contextual understanding, we expect BERT to outperform both traditional and standard deep learning models.
- **Model Optimization:** Hyperparameter tuning (using methods like grid search) and techniques such as dropout and regularization are applied to the deep learning models to prevent overfitting and further boost performance.

7. Multi-Label Classification

Given that toxic comments may belong to more than one category (e.g., a comment can be both toxic and obscene), the model is designed to handle multi-label classification. The architecture is extended to allow for multiple output labels per input, and binary cross-entropy loss is used to account for the multi-label nature of the problem.

8. Handling Class Imbalance

To mitigate the issue of imbalanced data, several techniques are employed:

- Oversampling the minority class.
- Undersampling the majority class.

Class-weighted loss functions, which assign higher penalties for misclassifying minority class examples. These techniques help ensure that the model is capable of detecting toxic comments despite their relatively lower frequency in real-world datasets.

9. Deployment Considerations

Once an optimal model is identified, the final step involves preparing the model for deployment. The model can be integrated into a real-time system for social media content moderation. Techniques such as batch processing and real-time API integration are considered to ensure that the model

© 2024 IJSRET

performs efficiently at scale, without compromising speed or accuracy.

10. Interpretability and Fairness

To ensure that the model is interpretable and fair, explainability techniques such as LIME (Local Interpretable Model-agnostic Explanations) or SHAP (SHapley Additive exPlanations) are applied to understand which features (words or phrases) contribute most to a prediction. Additionally, the model is evaluated for potential biases, ensuring that it does not disproportionately misclassify comments from particular groups based on factors such as race, gender, or ethnicity.

V. CONCLUSION

In this paper, we have presented a comprehensive approach to classifying toxic comments on social media using machine learning techniques. The increasing volume of toxic content, including hate speech, threats, and offensive language, has made automated moderation essential to ensure safer online environments. Through the application of traditional machine learning algorithms like logistic regression and SVM, as well as advanced deep learning models such as RNNs, CNNs, and BERT, we demonstrated the potential for effectively detecting and categorizing toxic comments.

Our study showed that while traditional models provide a baseline for classification, deep learning models, particularly transformer-based architectures like BERT, outperform them significantly in terms of accuracy, precision, and recall. This is due to their ability to capture the contextual nuances of language, which is essential for identifying complex forms of toxicity. Additionally, by employing techniques such as TF-IDF, word embeddings, and BERT-based contextual embeddings, we enhanced the feature extraction process, further improving model performance.

Handling challenges such as imbalanced datasets and multi-label classification through methods like SMOTE and class-weighted loss functions has enabled our models to more effectively detect toxicity across various categories. Furthermore, deploying explainability techniques ensures that the models are interpretable and fair, reducing the risk of biased or unfair content moderation.

In conclusion, machine learning and deep learning approaches, particularly transformers, provide a powerful solution for the automated classification of toxic comments. This work contributes to the ongoing efforts to create safer and more respectful online communities, offering a scalable and effective tool for content moderation on social media platforms. Future work may involve expanding the dataset, exploring additional models, and refining techniques to further enhance performance and reduce bias.

REFERENCES

1. H. M. Saleem, K. P. Dillon, S. Benesch, and D. Rut hs, “A Web of Hate: Tackling Hateful Speech in Online Social Spaces,” 2017, [Online]. Available: http:// arxiv.org/abs/1709.10159.
2. M. Duggan, “ Online harassment 2017,” Pew Res., pp. 1 –85, 2017, doi:202.419.4372.
3. M. A. Walker, P . Anand, J. E. F. Tree, R. Abbott , and J. King, “ A corpus for research on deliberation and debate,” Proc. 8t h Int . Conf. Lang. Resour. Eval. Lr. 2012, pp. 812–817, 2012.
4. J. Cheng, C. Danescu-Niculescu-Mizil, and J. Leskovec, “ Antisocial behavior in online discussion communities,” Proc. 9t h Int . Conf. Web Soc. Media, ICWSM 2015, pp. 61–70, 2015.
5. B. Mathew et al., “Thou shalt not hate: Countering online hate speech,” Proc. 13th Int. Conf. Web Soc. Media, ICWSM 2019, no. August , pp.369–380, 2019.
6. C. Nobata, J. Tet result , A. Thomas, Y. Mehdad, and Y. Chang,“ Abusive language detection in online user content ,” 25t h Int . World Wide Web Conf. WWW 2016, pp. 145–153, 2016, doi:10.1145/2872427.2883062.
7. E. K. Ikonomakis, S. Kotsiantis, and V. T ampakas, “Text Classification Using Machine Learning Techniques,” no. August , 2005.
8. M. R. Murthy, J. V. . Murt hy, and P . Reddy PVGD, “ Text Document Classification based on Least Square Support Vector Machines with Singular Value Decomposition,” Int. J. Comput . Appl., vol. 27, no. 7, pp. 21–26, 2011, doi: 10.5120/3312-4540.
9. E. Wulczyn, N. Thain, and L. Dixon, “ Ex machina: Personal attacks seen at scale,” 26th Int. World Wide Web Conf. WWW 2017, pp. 1391–1399, 2017, doi: 10.1145/3038912.3052591.
10. H. Hosseini, S. Kannan, B. Zhang, and R. Poovendran, “ Deceiving Google’s Perspective API Built for Detecting Toxic Comment s,” 2017, [Online]. Available: http:// arxiv.org/abs/1702.08138.
11. Y. Kim, “ Convolutional neural networks for sentence classification,” EMNLP 2014 - 2014 Conf. Empire. Methods Nat . Lang. Process. Proc. Conf., pp. 1746–1751, 2014, doi: 10.3115/v1/d14-1181.
12. R. Johnson and T . Zhang, “ Effective use of word order for text categorization with convolutional neural networks,” NAACL HLT 2015

© 2024 IJSRET