Multi-Label Abusive Comments Identification through Machine and Deep Learning Approaches!

Abstract — Our research explores the complex triggers of toxicity in online conversations, revealing how even seemingly harmless comments can spark intense hostility. Given the significant impact that social media opinions have on individuals, addressing online toxicity is crucial. To tackle this issue, we introduce a sophisticated multi-label classification framework that combines TF-IDF and Word2Vec techniques for comprehensive vectorization. This framework integrates basic textual data with advanced metrics from previous studies, enabling detailed monitoring of sentiment changes, topic trends, and conversational context.

By utilizing a variety of algorithms, including Logistic Regression, AdaBoost, Naive Bayes, Gradient Boosting, and Neural Network architectures such as LSTM and Bi-LSTM, our model effectively identifies four distinct types of toxicity: 'toxic', 'obscene', 'insult', and 'non-toxic'. Our study highlights the importance of understanding contextual nuances and sentiment variations in online interactions. It advocates for the use of advanced natural language processing techniques to promote constructive discourse and enhance digital engagement.

Moreover, our research emphasizes the ever-changing nature of online conversations, stressing the need for adaptable frameworks that can capture and respond to evolving patterns of toxicity.

Keywords—***Toxicity, Multi-Label Classification, TF-IDF, Word2Vec, Machine Learning, LSTM, Bi-LSTM, Deep Learning***

I. INTRODUCTION

The rise of social media platforms like Instagram, Facebook, and YouTube has revolutionized how people connect and form communities. However, this increased connectivity has also introduced significant challenges in managing content to ensure respectful interactions. The surge in online activity has led to a rise in toxic comments—such as abuse and rude behavior—that erode healthy dialogue and foster distrust, negatively affecting user satisfaction and potentially driving users away. As the number of users grows, effectively moderating discussions to prevent incivility becomes increasingly challenging, impacting the overall quality of online interactions.

Online toxicity, including bullying and violence, poses serious problems for digital communication. To address these issues, strategies such as counterspeech, community guidelines, and automated detection systems are essential. Given that toxic content can quickly escalate and one harmful comment can trigger further negativity, it's crucial to understand what causes such behavior. Identifying these triggers is key to developing interventions that prevent online harassment from worsening.

Unchecked online toxicity has severe repercussions, including psychological distress, suicidal thoughts, extreme social isolation, reputational damage, and even threats of violence. Research has highlighted a significant gap in understanding the root causes of toxic interactions. To develop effective strategies for mitigating these harmful behaviors, it is essential to explore the triggers that initiate toxic dialogues. This research is vital for protecting individuals and preventing a decline in user engagement on platforms.

Developing effective systems to detect and manage harmful content is crucial for maintaining the quality of online discussions. Advances in natural language processing (NLP) and machine learning have led to automated systems that can identify toxic comments. However, these systems often lack a deep understanding of the underlying causes of toxic responses, which is necessary for creating preventive solutions.

Our research aims to improve digital environments by leveraging advanced NLP and machine learning techniques. We are developing a robust system designed to detect and address toxic content before it reaches the broader community. This proactive approach focuses on preventing the escalation of harmful interactions by tackling them early. By using sophisticated algorithms, such as machine learning models and neural networks, along with detailed textual analysis, we aim to understand the nuances of toxic comments. This understanding is crucial for designing effective interventions that can automatically filter out toxic content and ensure the safety and integrity of online conversations.

Our approach offers a scalable solution to the growing issue of online toxicity, promoting a safer and more inclusive digital space. We seek to create a comprehensive framework that not only detects but also prevents the recurrence of harmful behaviors, enhancing online platform moderation and improving user experience.

II. DATASETS INFORMATION

The dataset utilised here is sourced from Kaggle and comprises comments extracted from Wikipedia talk page edits. Human raters have manually annotated these comments to identify toxic behaviour. This dataset encompasses four toxicity categories: 'toxic', 'obscene', 'insult', and 'non-toxic'. Each category is represented by boolean values (0 or 1) indicating the presence or absence of the respective type of toxicity. Additionally, we have introduced an additional 'non-toxic' label based on the absence of any toxic behaviours,

contributing to a balanced dataset suitable for classification purposes.

A. *Preprocessing*

During pre-processing, the textual data underwent critical procedures to enhance its quality for analysis. Initially, text was tokenized to separate words, then converted to lowercase for uniformity. Lemmatization reduced words to basic forms, standardizing and reducing dimensionality. Stopwords were removed to prioritize meaningful content, and refined further from lemmatized tokens. Additionally, statistics like total tokens, sentences, and punctuation marks provided insight. Overall, these methods ensured clean, standardized data, primed for analysis. The generated data frame was used to generate the TF-IDF matrix and Word2Vec embeddings separately.

1) **TF-IDF**

TF-IDF (Term Frequency-Inverse Document Frequency) is a popular method used in natural language processing to convert text data into numerical representations suitable for machine learning. It calculates the importance of each word by comparing its frequency in a document with its frequency across all documents in the dataset. This approach helps identify words that are unique and relevant to specific documents while reducing the importance of common words like 'the' or 'and'. In our research, TF-IDF vectors are instrumental in extracting meaningful features for predicting toxicity labels such as 'toxic', 'non-toxic', 'obscene', and 'insult'. This method enables the characterization of each document's textual content, facilitating accurate modeling of toxicity prediction tasks.

2) **Word2Vec**

Word2Vec is a widely-used technique in natural language processing that transforms words into dense vectors within a continuous vector space. This method helps represent words with similar meanings close together in the vector space. In our research, we utilize the Word2Vec model to capture semantic word relationships in textual data. By training Word2Vec on a large text corpus, it learns to assign numerical vectors to words based on their contexts. These embeddings contain semantic information that aids in analyzing word associations, identifying similarities, and supporting tasks like text classification and sentiment analysis. In our study, Word2Vec embeddings enhance the understanding of textual data and improve the performance of models used to predict toxicity labels such as 'toxic', 'non-toxic', 'obscene', and 'insult'. This approach enables the model to capture subtle semantic meanings inherent in the text, leading to more precise toxicity predictions.

III. METHODOLOGY

Our methodology involves categorising content using four different labels. To do this, we employed a robust architecture that integrates two essential embedding techniques: TF-IDF and Word2Vec. These embeddings provide the framework for capturing semantic and contextual information in text data. Furthermore, we employed a number of machine learning and deep learning techniques such as Logistic regression , Naive Bayes , Adaboost, LSTM, Bi-LSTM and Gradient Boosting models. This comprehensive approach allows us to reliably estimate the toxicity of text samples from numerous categories.

4.A. DEFINITIONS

1. Logistic Regression :

I evaluated our textual data using logistic regression, a well-known and interpretable categorization technique. While originally designed for binary classification, we extended it to the multinomial logistic regression variation to handle multi-class classification tasks effectively. Logistic regression calculates the likelihood of each class based on input features and provides insights through its coefficients. By combining logistic regression with word embedding approaches such as Word2Vec and TF-IDF, we achieved successful text classification into four separate categories.

2. Naive Bayes

I employed Naive Bayes, a simple but effective probabilistic classification algorithm, to categorize textual input into four unique labels. Prior to training the model, I conducted extensive text preprocessing and utilized both TF-IDF and Word2Vec methods to convert the text into numerical features suitable for Naive Bayes classification. The trained classifier used Bayes' theorem and the assumption of feature independence to compute the conditional probabilities for each class based on its intrinsic properties. This approach demonstrated the algorithm's stability across different feature representation methods.

3. Ada Boost

I used AdaBoost, an ensemble learning algorithm, which sequentially trains weak classifiers on weighted data, adjusting weights based on classification accuracy to prioritize misclassified examples. By employing decision stumps as weak learners, AdaBoost aggregates their predictions through weighted averaging to produce a final classification. Preprocessing was carried out to ensure compatibility with AdaBoost, especially in tasks requiring numerical features like TF-IDF or Word2Vec embeddings. This approach highlighted AdaBoost's effectiveness in leveraging different feature representations.

4. Gradient Boosting

I employed Gradient Boosting, a versatile ensemble method, to analyse the textual data. Originally designed for regression, it has been extended for classification by iteratively creating decision trees to minimize loss functions. Although commonly used for binary classification, it can be adapted for multi-class tasks. By combining it with Word2Vec and TF-IDF embeddings, I efficiently categorized text into four classes, leveraging its interpretability and ability to handle complex data relationships.

5. Long Short Term Memory (LSTM)

I utilized an LSTM model to predict multiple toxicity labels ('toxic', 'non-toxic', 'obscene', 'insult') using a two-layer architecture with dropout regularization. The input data, represented as TF-IDF vectors, was reshaped for LSTM processing and optimized with binary cross-entropy loss and the Adam optimizer over 10 epochs. Additionally, Word2Vec embeddings, which capture semantic relationships and language nuances by positioning similar words close in a multidimensional space, were used to enhance context understanding for toxicity prediction. While TF-IDF provides valuable contextual depth, Word2Vec offers improved

accuracy by better capturing semantic nuances in complex tasks.

6. Bi-Long Short Term Memory (Bi-LSTM)

The BI-LSTM model uses two bidirectional LSTM layers (64 and 32 units) with dropout (0.3) to predict four toxicity labels ('toxic', 'obscene', 'insult', 'non-toxic') simultaneously. The input data is converted into TF-IDF vectors and Word2Vec embeddings to capture word relationships and semantic meanings. During training, binary cross-entropy loss and the Adam optimizer are used over 10 epochs with a batch size of 32, alongside validation data to monitor and prevent overfitting. Sigmoid activation facilitates multi-label classification. For Word2Vec, training employs binary cross-entropy and the Adam optimizer over 15 epochs with the same batch size. The model demonstrates robust performance in toxicity prediction with varying accuracy across different embeddings. Both the Bi-LSTM and LSTM models show similar performance when trained on TF-IDF vectors, yielding comparable results in toxicity label prediction tasks.

IV. Proposed Model

The framework shown in Figure 1 depicts a thorough approach to categorizing text into four distinct categories: poisonous, obscene, insult, and nontoxic. This method combines established techniques like TF-IDF (Term Frequency-Inverse Document Frequency) with cutting-edge methodology like Word2Vec embeddings, machine learning, and deep learning algorithms. The steps are as follows.

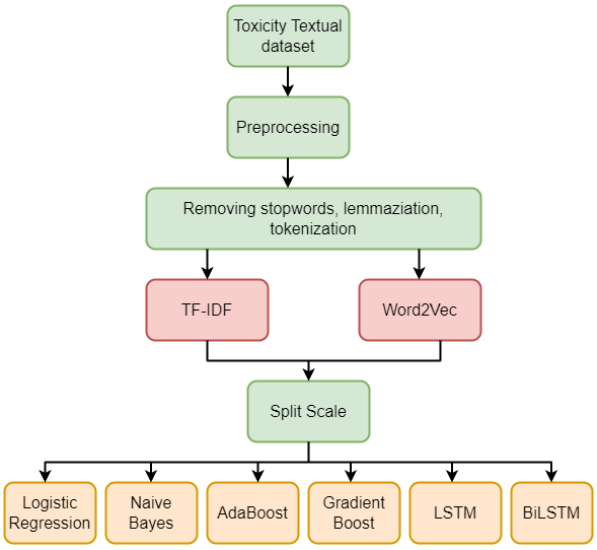

Fig.1. A snippet of the proposed approach procedure

Initially, a diversified dataset with classifications such as toxic, obscene, insult and non-toxic was gathered. The acquired data was then preprocessed, including stop word removal and punctuation cleanup, to improve its quality and relevancy. Following preprocessing, two different strategies for feature extraction were used: TF-IDF transformation and Word2Vec embeddings. The TF-IDF transformed data was then classified using a variety of machine learning techniques such as AdaBoost, Logistic Regression, Gradient Boosting, and Naive Bayes. Similarly, the Word2Vec embeddings were combined with the same machine learning methods for categorization. In addition, deep learning methods, notably Long Short-Term Memory (LSTM) and Bidirectional LSTM (BiLSTM), were used with both TF-IDF converted data and Word2Vec embeddings to capture detailed patterns and dependencies in text. Throughout the process, feature scaling was used to ensure consistency and optimal performance across all algorithms. Finally, the trained models were tested in terms of accuracy, precision, recall, and F1-score to determine their usefulness in categorising text based on toxicity and content.

V. Results

In our project, we have employed various evaluation metrics to measure the effectiveness of the deep learning and ML models classify toxicity in social media. To measure the effectiveness of the algorithms, we considered multiple performance metrics, each providing unique insights into the models' capabilities to accurately predict toxicity in comments. These metrics include:

Test Accuracy: To evaluate a model's overall performance, divide i.e the number of properly predicted cases by the total number of instances.

$$Accuracy(a) = \frac{K + L}{K + L + M + N}$$

Precision: Precision refers to how accurately a model predicts good outcomes. It is derived by dividing the number of true positives by the sum of true positives and false positives, representing the proportion of correctly detected positives out of all positive predictions.

$$Precision(p) = \frac{K}{K + M}$$

Recall: It assesses the model's ability to identify all of the appropriate instances from a given group. You calculate it by dividing the number of correctly detected positives by the total number of positive instances.

$$recall(r) = \frac{K}{K + N}$$

F1-Score: It's a single score that shows how good the model is overall. It combines precision and recall into one number, giving an idea of how well the model deals with mistakes.

$$F1 - Score(r) = \frac{2(p * r)}{p + r}$$

In the equations, K, L, M, and N represent different outcomes of classification. K is the number of true positives (correct positive predictions), L is the number of true negatives (correct negative predictions), M is the number of false positives (incorrect positive predictions), and N is the number of false negatives (incorrect negative predictions).