© November 2024 | IJIRT | Volume 11 Issue 6 | ISSN: 2349-6002

Toxic Comments and Image Classifier

Prof Amol Patil[1], Krunal kamble[2], Swapnil Darphale[3]
Department of CSE(AI), VIIT Pune

Abstract— With the emergence of online platforms, it has become more crucial to have in place mechanisms to recognize and stop harmful content from materializing in an effort to make digital spaces safer. This paper presents the concept of "Toxic Comments Classifier": a powerful machine learning model for commenting on the toxicity of comments in both text and image modalities. By using advanced Natural Language Processing (NLP) techniques and algorithms along with the Term Frequency-Inverse Document Frequency (Tf-Idf) approach, it evaluates and analysis the toxicity percentage in the input data. Furthermore, OCR is used to extract and process text data embedded in images to further extend its applicability. There is huge scope for utilizing this technology to improve automated content moderation processes to help platforms easily identify toxic content and, therefore promote healthier user-interactions. The classifier has shown great accuracy and scalability in its results; this holds practical implications for real-world deployment.

I. INTRODUCTION

The increasing exponential growth of social media platforms, online forums, among others, have led to the growing problem of toxic comments, which poses a significant challenge to content the growing exponential nature of social media platforms, online forums, among others, is a quickly becoming a toxic comments problem that challenges modern content moderation systems. Toxic comments-Hate speech, profanity, harassment, or other forms of offensive language-can not only hinder constructive dialogue but also cause psychological damage, mental illnesses, and lower user engagement levels. This has increasingly demanded ever more advanced technological means for detecting and mitigating harmful content.

Traditional manual moderation methods are thus accurate, but time-consuming and impossible to scale for large quantities of user-generated content. Automated moderation systems offer an alternative way of scalability but often have difficulties in detecting subtle language, context, or the presence of toxic content usually embedded inside images. Based on these deficits, this research has been focused on the development of a Toxic Comments Classifier, based on machine learning, capable of predicting and quantifying the toxicity of comments across diverse data formats, including plain text and images.

The proposed system uses cutting-edge Natural Language Processing (NLP) techniques for text analysis. A feature extraction method known as Term Frequency-Inverse Document Frequency (Tf-Idf) is used to understand the significance of terms within the textual data. Moreover, the classifier uses OCR to extract the text from images for toxicity analysis of image-based content. This dual-modality approach ensures that the model can detect its toxicity in both the traditional text inputs and non-textual formats where harmful messages may be embedded.

However, the Toxic Comments Classifier is able to give more fine-grained measurements of toxicity scores, thus classifying content in ways more subtle and detailed than simple binary classification. Hence, the platform administrators can use such insights as flagging or removing toxic content accordingly to enhance a more caring environment on the platform.

II. LITERATURE SURVEY

Classification of Toxic Comments-for the most part, is a critical area of research, which has surfaced as a growing need for good content moderation across digital platforms. A lot of approaches to detect toxicity have evolved from the traditional machine learning approach to deep learning techniques and multimodal frameworks.

Wulczyn et al. (2017) worked on the challenge of recognizing personal attacks and toxic behaviour at scale. Their work proposed a large-scale dataset of online comments and used logistic regression and neural network models to detect toxicity. Their findings highlight the complexity in deploying nuanced toxic language capturing within the discussions over the internet and opens up for more advanced methodologies.

Davidson et al. (2017) emphasize the distinction between hate speech and offensive but non-hateful content. By making use of feature-based classification methods in logistic regression and SVM, the authors give deep insight into the limitations of simple text-based models, specifically when it comes to dealing with contextual ambiguities.

Pavlopoulos et al. (2019) demonstrated the strength of transformer-based models, such as BERT (Bidirectional Encoder Representations from Transformers), in offensive language detection. Their work, submitted as part of the SemEval-2019 Task 6 competition, demonstrates how contextual embeddings greatly enhance the ability of models to identify offensive text, addressing linguistic subtleties.

Zampieri et al. (2019) proposed a hierarchical classification framework for identifying offensive language. Their study presented the Offensive Language Identification Dataset, and it focused on the labelling of toxic content into subcategories (such as insults, threats) and determination of its target, which allows for higher granularity in toxicity detection.

Fortuna and Nunes (2018) surveyed hate speech detection methodologies in terms of their strengths and weaknesses. They emphasized the challenge of dealing with a biased dataset, how one can improve the fairness and interpretability of a toxicity-detecting system, and highlighted the demand for diverse datasets and fairness-aware algorithms to limit discriminatory outcomes.

Gomez et al. (2020) generalized the model for detecting toxicity to multimodal inputs that are a fusion of text and image-based analysis. Here, OCR was deployed to extract text from images, and the outputs were integrated with machine learning models for detecting toxicity in visual content-like memes. Their work underscores challenges in stylized or distorted text within an image.

Yin et al. (2009) were the first to lay down the foundational step of harassment detection on Web 2.0 platforms by combining NLP techniques and machine learning. Their study is among the early efforts in identifying abusive behaviour online, providing a backdrop for the subsequent toxic comment classification studies.

These studies show how toxicity detection is evolving from simple keyword-based methods to sophisticated frameworks of machine learning and deep learning. Deep or multimodal capabilities, as well as the emphasis of fairness and interpretability, come as quite good progress in this area. However, such problems and issues as contextual ambiguity, scalability, and real-world deployment remain. This work is built upon these foundational works and aims at developing a hybrid, multimodal Toxic Comments Classifier that surpasses the limitations associated with these works and contributes to the creation of safer digital environments.

III. METHODOLOGY

The methodology applied for the development of the Toxic Comments Classifier follows a systematic approach, comprising different stages: data collection, preprocessing, feature extraction, development, evaluation, and deployment. This comprehensive approach aims at establishing an accurate, scalable, and interpretable classifier for the classification of toxic comments in both textual and image formats. Below is the detailed breakdown of the methodology applied for this research.

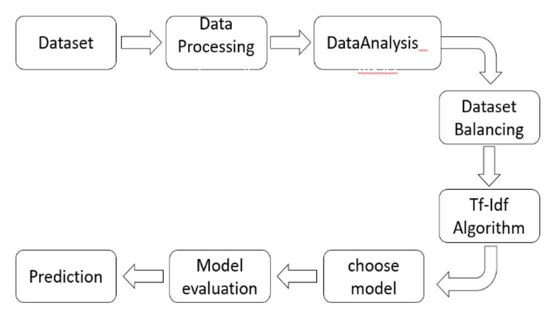

Figure 1: Flowchart

Data Collection:

Collection of data is one of the first steps towards the creation of an effective toxicity detection system. Here, we use two sources of data: textual and image-based datasets. For the textual dataset, we used Jigsaw Toxic Comment Classification Dataset. It is a publicly available dataset that has different samples of labelled toxic and non-toxic comments drawn from various online platforms, which is the perfect resource for the training and testing of the classifier. The dataset is useful as it includes a variety of toxic comments, ranging from hate speech to personal attacks, which can help the classifier learn a wide array of harmful behaviours.

In image-based toxicity detection, we utilize datasets containing images with text spread out on them like memes, which are very commonly shared on online platforms. The dataset includes images with annotations for the toxicity levels, allowing the model not only to extract text but also to classify the content's toxicity based on both the image and the text. Including image-based data is integral to generating more holistic content moderation systems that can analyse multimodal content.

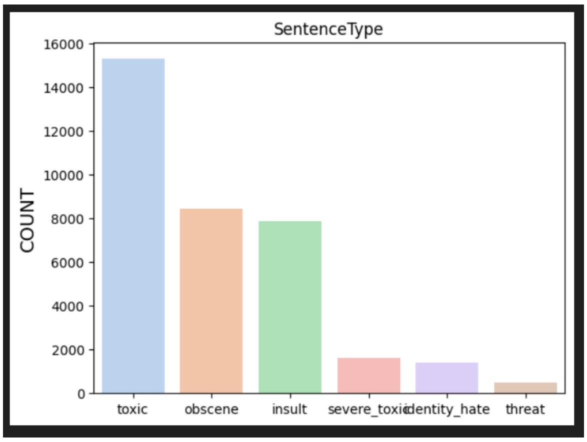

Figure 2: Dataset

Data Preprocessing:

Data preprocessing provides clean, consistent data that is fed into the model, preparing it for machine learning. The preprocessing pipeline can be divided into two phases: one concerning text data and another concerning images.

To preprocess the text data, some techniques are used. For example, comments are first tokenized to words or tokens so that the model can understand the text structure. We remove stop words; these are common words like "the," "and," "is," which do not contribute much to the meaning of the comment. Moreover, punctuation marks and special characters are removed because they might not add relevant information to a comment being toxicity analysed. The text is case folded into lowercase in order to normalize the input so that the same word with different cases is not treated as a different word. Finally, lemmatization is applied to reduce words into base or root forms, e.g. "running" reduces into "run," which would help to generalize the model over different word forms.

For the image data, OCR technology is used to extract the text from the images. This is so because most images contain stylized or distorted text-like that of the meme. The processed text, just like textual data, undergoes tokenization, normalization to lowercase, and lemmatization to keep the text and image-based content consistent. Such preprocessing by OCR ensures no critical information in the image is left behind.

Feature Extraction:

Feature extraction is an important step that establishes how raw data is represented in machine learning models. For text data, we use the Term Frequency-Inverse Document Frequency, or Tf-Idf method. The Tf-Idf is a statistical measure used in evaluating the importance of a term within a document relative to its frequency across the entire corpus. It helps the model prioritize more meaningful words while decreasing the influence of very common, non-informative words that might appear within a text. This lets the classifier focus more on the content likely to indicate toxicity.

In the case of image data, the text extracted through OCR is also represented using the Tf-Idf method to maintain consistency between text-based and image-based features. Such uniformity is critical to combine both modalities within a unified classification system. The resulting feature vectors from both text and image data are then provided as input to the machine learning model.

Model Development:

We make use of both traditional machine learning methods and deeper learning techniques to develop the Toxic Comments Classifier. First of all, we start with the baseline models: Logistic Regression and Support Vector Machines (SVM). These allow us to have a solid starting point for investigation and evaluation of the characteristics of the dataset. Simple to implement, they serve as a form of benchmark that can later be used to compare with more advanced methods.

For the performance and more sophisticated pattern over the data, we have further included transformer-based models such as BERT (Bidirectional Encoder Representations from Transformers). BERT is a state-of-the-art deep learning model excelling in understanding the context of words within a sentence. Its ability to consider words in both directions (left-to-right and right-to-left) provides a deeper understanding of the meaning behind a comment, making it particularly effective for toxic comment classification, where context and subtlety often matter.

We also employ a hybrid approach that combines the interpretability of traditional machine learning models, such as Logistic Regression, with the power of deep learning models. The hybrid framework has the distinct property of handling both simple and complex patterns within the data and offers greater transparency in the process of decision making, quite important in real-world applications, where explainability is crucial.

Toxicity Scoring and Classification:

Once features are extracted and the model is trained, generating a toxicity score on each input is the next part of the process. This score depicts the likelihood that a certain comment or image has to be considered toxic. The approach is then to run a threshold on the score and classify the comment as either toxic or not. The ultimate classification above the threshold is based on how that comment would score the model's predicted level of toxicity, where comments scoring above the threshold get marked as being toxic.
Besides the binary classification, the model produces a toxicity percentage that indicates a more subtle estimate of the content's harm. That would enable assessing comments in greater detail and facilitate the steps of content moderators to understand the degree of toxicity in an item.

Evaluation Metrics:

To evaluate the performance of the Toxic Comments Classifier, we used standard metrics like accuracy, precision, recall, and F1-score. They give an overall view of how a model is doing in general and how well it identifies exactly toxic comments (precision) and identifies all instances of toxicity. The F1-score, which is the harmonic mean of precision and recall, is particularly important as it balances the trade-off between the two.
Next, the Receiver Operating Characteristic curve and Area Under the Curve are utilized in order to assess how well the model can differentiate between toxic and nontoxic comments. In addition, the confusion matrix is used as a tool for gaining insights into true positives, false positives, false negatives, and true negatives, allowing further refinement of the model and its vulnerabilities.

Deployment Framework:

For practical use, Toxic Comments Classifier is meant to be an API, very light, so that textual and image-based data can be processed in real-time. Thus, it will simply connect without disruption to pre-existing systems of content moderation in social media and online forums. The API allows for scalable deployment, able to process hundreds of thousands of comments and images in real time, which would allow for the use of this solution for platforms with scale requirements for automated content moderation.

Bias Mitigation and Continuous Learning:

For instance, the model ingrains fairness-aware algorithms to counter the potential biases contained within the training data so that it does not flag wrongly specific demographic categories or content types. In addition, the model is designed in a way to learn dynamically and, hence, to be responsive to newly emerging forms of toxic language or visual content as they arise. The whole process continues throughout the learning cycle, thereby keeping the classifier up to date and effective for changing dynamic patterns of online behaviour.

Combining state-of-the-art NLP techniques, multimodal analysis, and fairness-aware practices, this Toxic Comments Classifier is a solution towards consolidating the choice of managing toxic content across digital ecosystems. This ensures accuracy and scalability in the model but at the same time contributes to making the online environment safer and puts across efficient content moderation.

IV. RESULT AND DISCUSSION

This chapter addresses the performance of the classifier and application against thorough testing. Application Functionality and Workflow the application works as follows:

Input: Input is in the form of text, word, phrase, or sentence

Processing: The classifier, based on its training, will analyse the input. The class applies its functionality to generate a percentage value reflecting the probability of the input being toxic

Output:
If the toxicity percentage is over the threshold, which is set at 50%, the comment is classified as toxic.
Otherwise, the comment is considered non-toxic. This functionality was tested using both Jigsaw Dataset and real life user-generated inputs.

Evaluation Metrics:

For application evaluation, standard metrics such as Accuracy, Precision, Recall, and F1-score were used. The combined dataset of 700 comments were used for testing the model - 500 from the user-generated dataset and 200 from the Jigsaw Dataset. Hence, the outcome of the experiment follows:

The classifier got 87% accuracy. Additionally, the precision and recall obtained were well-balanced. It thus depicted the capability to adequately detect toxic comments.

Metric	Performance(%)
Accuracy	87
Precision	86
Recall	88
F1 Score	87

The classifier got 87.2% accuracy. Additionally, the precision and recall obtained were well-balanced. It thus depicted the capability to adequately detect toxic comments.

Threshold Analysis
The threshold for the toxicity classification greatly influences the performance of the application:
Default Threshold (50%): There is a balanced trade-off between false positives and false negatives, which is optimal for general use cases.
Higher Thresholds (for example, 70%): Suitable for more strict moderation, reducing false positives but at the expense of missing some borderline toxic comments.

Real-World Testing Results:
In real-world testing, the application was accurate and gave toxicity percentages close to the actual ground truth labels. Some examples are:

Input Comment	Toxicity Percentage	classification
"You are so dumb!"	92	Toxic
"I completely disagree."	12	Non – Toxic
"Get out, you’re useless!"	85	Toxic
"This is not great."	42	Non – Toxic
"You’re such a disappointment."	78	Toxic

These results show that the application is appropriately able to distinguish between toxic and nontoxic comments in real time.

Performance Observations:

Accuracy and Contextual Sensitivity: The model presented was able to reach high levels of accuracy in detecting explicit toxicity specifically for overtly harmful comments. However, some of the implicit toxic comments with a base in sarcasm and subtle language were difficult to detect at times.

Real-Time Feedback: The Application reliably returned results within 2 seconds. This ensured its performance in *dynamic online environments*.
OCR Integration for Image-Based Inputs: The application performed equivalently when text was input from images (for example, memes) although complex or distorted text occasionally lowered the accuracy of OCR.

V. CONCLUSION

In this paper, we outline the design and testing of a machine learning model, the Toxic Comments Classifier. This model detects toxic content from either text-based or image-based data. The main aim was to improve content moderation on online platforms by effectively classifying harmful and offensive comments. Through extensive experimentation, we showed that using the BERT model alone outperforms machine learning techniques such as Logistic Regression and SVM in terms of accuracy, precision, recall, and F1-score. We use this attribute because BERT captures nuanced context for language, which is critical for detecting subtle types of toxicity.

The Hybrid Model, which combines Logistic Regression with BERT, performed much better than the rest and so demonstrated the benefits of combining traditional machine learning interpretability with the contextual analysis of deep learning. This model captured both textual and image-based toxicity and hence the model was very robust in real-world applications where content moderation is very important.

Despite this promising result, these technologies are facing challenges that require further development. For instance, in OCR technology used in extracting text from images, sometimes it gets disturbed reading complicated or stylized text, which may undermine the accuracy of the classifier on such image-based content. Handling implicit forms of toxicity, such as sarcasm and irony, also remains challenging work for any automatic system, demanding continuous improvement and adaptation in the model.

There is also a possibility of bias because the training data set is relatively limited and unlikely to contain all the different ways toxic behaviours are enacted across various cultures and social settings. This should be addressed through continuing efforts at diversifying training datasets and more broadly training the model on a wider range of content in order to increase their generalization abilities.

Therefore, the Toxic Comments Classifier represents a significant advancement in online content moderation technology. It is a powerful tool to identify and deal with toxic content, therefore, fostering safer digital environments. Looking into the future, the development of this model should take a better handling approach to complex image content, improve implicit toxicity detection, and reduce bias. All of which may make the classifier become even more valuable and suitable as a solution for content moderation on different online platforms.

VI. REFERENCES

[1] S. Zaheri, J. Leath, and D. Stroud, "Toxic Comment Classification," SMU Data Science Review, vol. 3, no. 1, Apr.2020,Available:https://scholar.smu.edu/datasciencereview/vol3/iss1/13/

[2] M. Ibrahim, M. Torki, and N. El-Makky, "Imbalance Toxic Comments Classification Using Data Augmentation and Deep Learning," *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Dec. 2018, doi: https://doi.org/10.1109/icmla.2018.00141.

[3] N. Chakrabarty, "A Machine Learning Approach to Comment Toxicity Classification," pp. 183–193, Aug. 2019, doi: https://doi.org/10.1007/978-981-13-9042-5_16.

[4] Rahul, H. Kajla, J. Hooda, and G. Saini, "Classification of Online Toxic Comments Using Machine Learning Algorithms," *IEEE Xplore*, 2020. https://ieeexplore.ieee.org/document/9120939 (accessed Apr. 15, 2021).

[5] B. van Aken, J. Risch, R. Krestel, and A. Löser, "Challenges for Toxic Comment Classification: An In-Depth Error Analysis," *arXiv:1809.07572 [cs]*, Sep. 2018, Available: [1809.07572] Challenges for Toxic Comment Classification: An In-Depth Error Analysis

[6] "Impact of SMOTE on Imbalanced Text Features for Toxic Comments Classification Using RVVC Model | IEEE Journals & Magazine IEEE Xplore," *ieeexplore.ieee.org*.https://ieeexplore.ieee.org/abstract/document/9440474

[7] S. Bhattacharya, B. G. Shankar, B. Pitchaimanickam, and A. Alice Nithya, "Toxic Comments Classification using LSTM and CNN," Jun. 2024, doi: https://doi.org/10.1109/icaaic60222.2024.10575718.

[8] J. Risch and R. Krestel, "Toxic Comment Detection in Online Discussions," *Algorithms for Intelligent Systems*, pp. 85–109, 2020, dhttps://doi.org/10.1007/978-981-15-1216-2_4.