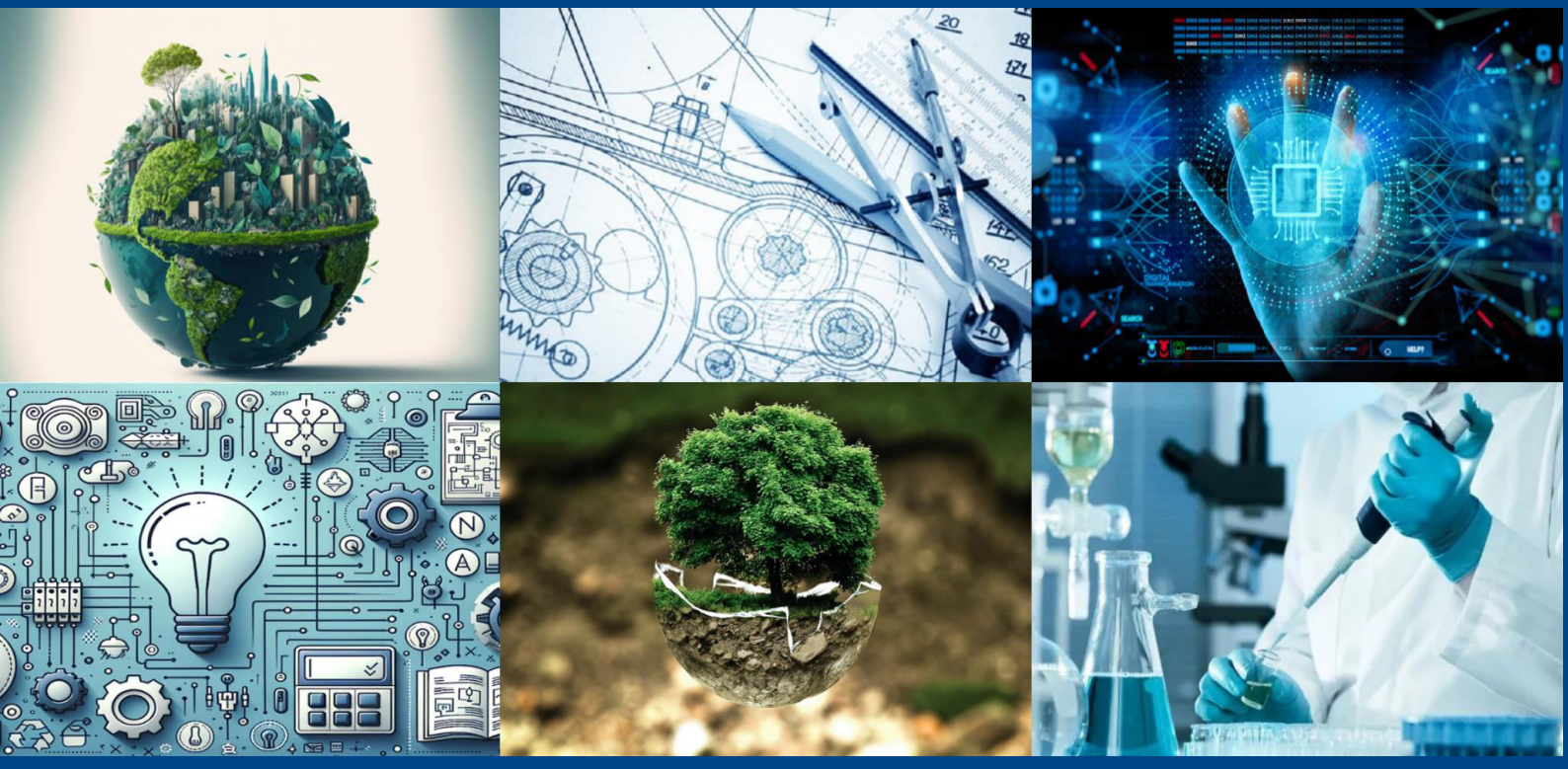
ISSN: 2582-7219

International Journal of Multidisciplinary Research in Science, Engineering and Technology

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.206 **Volume 8, Issue 5, May 2025**

© 2025 IJMRSET | Volume 8, Issue 5, May 2025| DOI:10.15680/IJMRSET.2025.0805213

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Toxic Comment Classification using NLP Techniques and Machine Learning Algorithms

G. Sireesha

Department of ECE, RVR&JC College of Engineering, Guntur, India

Ch. Indraja

Department of ECE, RVR&JC College of Engineering, Guntur, India

A. Venkata Praveen

Department of ECE, RVR&JC College of Engineering, Guntur, India

ABSTRACT: The explosion of digital content across social platforms has significantly increased the visibility of toxic, offensive, and harmful comments. Manual moderation is neither scalable nor consistent, necessitating intelligent, automated filtering systems. This study proposes a robust multi-label classification approach that leverages Natural Language Processing (NLP) and Machine Learning (ML) algorithms to detect and categorize toxic comments. The model operates using a One-vs-Rest classification strategy with TF-IDF vectorized inputs. The preprocessing pipeline includes normalization, stopword removal, and stemming to improve learning accuracy. Experimental evaluations were conducted using various ML classifiers including Logistic Regression, SVM, Random Forest, and others. Among these, the Support Vector Machine (SVM) demonstrated the highest F1-score and overall reliability. This framework has practical applications in moderating online discussions, enhancing user experience, and supporting safer digital communities.

KEYWORDS: Machine Learning Algorithms, Natural Language Processing Techniques, Toxic Comment Classification, Multi label Classification, TF-IDF, Accuracy, Precision, Recall, F1 Score, ROC Curve.

I. INTRODUCTION

The advent of interactive digital platforms such as social media, forums, and news portals has transformed online communication. However, these platforms are increasingly exploited to spread offensive, abusive, or threatening content that deteriorates public discourse and individual well-being. Managing such toxicity through human moderation is impractical at scale, motivating the development of automated tools to detect harmful content. Traditional text classification models often assume a one-to-one mapping between a comment and its category. In reality, a single comment may express multiple types of toxicity such as 'threat', 'insult', or 'obscene' language simultaneously. This paper addresses this complexity by formulating toxic comment classification as a multi-label classification problem, where each comment may be associated with several toxicity labels. The research aims to develop a comprehensive NLP-based ML framework to classify toxic comments. It involves meticulous text preprocessing, TF-IDF feature extraction, and the application of multiple machine learning algorithms using the One-vs-Rest strategy to handle the multi-label nature of the data.

II. LITERATURE REVIEW

Abhishek Aggarwal and Atul Tiwari [1] developed a multi-label classification model using traditional algorithms like Logistic Regression and Random Forest. Their findings highlighted the effectiveness of standard classifiers when combined with proper preprocessing and feature engineering for detecting multiple forms of toxicity.

Ozoh et al [2] emphasized the importance of preprocessing techniques such as tokenization, stopword removal, and vectorization for enhancing model performance. Their approach used standard ML models, reinforcing the significance of clean input data for accurate classification.

© 2025 IJMRSET | Volume 8, Issue 5, May 2025| DOI:10.15680/IJMRSET.2025.0805213

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Tsoumakas and Katakis [3] laid the foundation for multi-label classification methods by introducing strategies such as problem transformation and algorithm adaptation. These techniques are highly relevant to toxic comment detection, where a single comment may exhibit multiple toxic traits.

Mishra and Tripathi [4] explored the use of Convolutional Neural Networks (CNNs) for toxic language classification, demonstrating that deep learning models can learn semantic patterns more effectively than traditional algorithms in certain cases.
Further advancements include the use of transformer-based architectures like XLNet and BERT, which have shown significant improvements in understanding language context. Yang et al. (2018) demonstrated that autoregressive pretraining in XLNet enhanced the model's performance in text classification tasks.

Singh and Chand [5] explored multi-label text classification with a focus on evaluation metrics such as F1-score and precision, which are critical for imbalanced datasets like those used in toxicity detection.
Nobata et al. [6] introduced the use of linguistic features in combination with supervised learning for detecting abusive content on large-scale datasets, showing that handcrafted features can still be highly effective.

Murty et al. [7] proposed a hybrid model using Least Squares Support Vector Machines (LSSVM) and Singular Value Decomposition (SVD) to improve feature reduction and classification efficiency, particularly in high-dimensional textual data.

III. METHODOLOGY

The proposed framework for toxic comment classification follows a machine learning pipeline integrated with Natural Language Processing (NLP). It consists of the following stages:

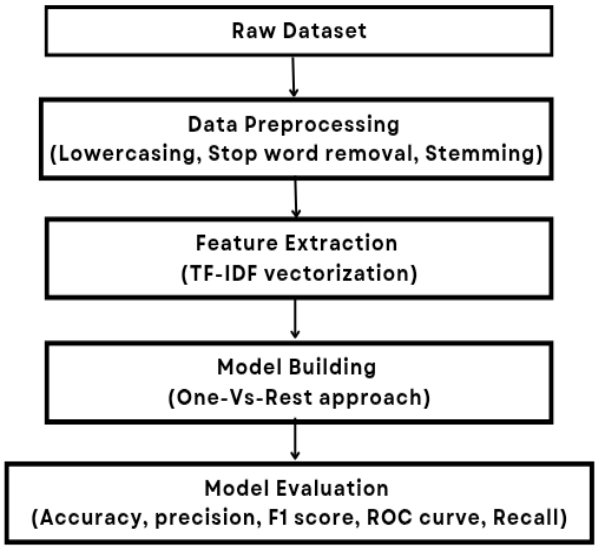


Fig. 1: Flow Chart

A. Raw Dataset

The dataset used in this project is sourced from a popular Kaggle competition focused on toxic language detection. It contains 159,571 user comments, each labelled with up to six binary categories: toxic, severe toxic, obscene, threat, insult, and identity hate.

B. Date Preprocessing Techniques

In this project, raw text data was first cleaned by converting to lowercase and removing punctuation, special characters, and stop words. Tokenization was performed to split text into individual words, followed by lemmatization to reduce words to their base forms.

© 2025 IJMRSET | Volume 8, Issue 5, May 2025| DOI:10.15680/IJMRSET.2025.0805213

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Fig. 2: Flow Chart for Data Preprocessing

1) Text Normalization (Lowercasing):

Text normalization in NLP involves converting raw text into a consistent format to reduce variations and improve model understanding. It helps focus on meaningful patterns by eliminating inconsistencies like case differences or spelling variations.

2) Stopword Removal:

Stopword removal is a key NLP preprocessing step that eliminates common words like "the," "is," and "and," which carry little semantic meaning. This helps reduce data size and focuses the model on more informative terms. In this project, stopwords were removed using NLTK's predefined English stopword list by filtering out these words after tokenizing each comment.

3) Stemming:

Stemming is an NLP technique that reduces words to their root form, helping to group different grammatical forms of a word under a single representation. This simplifies the text and lowers data dimensionality. For example, "connect," "connected," and "connection" are all reduced to a common stem.

$$\text{Stem}(w) = W_{root}$$

Where:
w is the original word (e.g. "playing")
Wroot is the stemmed form (e.g., "play")

C. Feature Extraction

Feature extraction is an essential step in NLP, converting cleaned text into numerical form that machine learning models can understand. In this project, TF-IDF (Term Frequency–Inverse Document Frequency) was used to represent the importance of words in a comment relative to the entire dataset. Unlike simple word counts, TF-IDF reduces the weight of common words and highlights more meaningful ones, helping the model focus on key terms. It balances how often a word appears in a document with how rare it is across all documents.

$$\text{TF-IDF}(t, d, D) = \text{TF}(t, d) \times \text{IDF}(t, D)$$

Where:
t is a term (word)
d is a specific document
D is the entire set of documents (corpus)

1.Term Frequency (TF): Measures how frequently a word t appears in a single document

$$\text{TF}(t, d) = \frac{f\,t, d}{\sum_k f\,k, d}$$

2. Inverse Document Frequency (IDF): Reduces the weight of common words and increases the weight of rare ones.

$$\text{IDF}(t, D) = \frac{N}{1 + |\{d \in D : t \in d\}|}$$

D. Model Building

Six supervised machine learning algorithms were explored using a One-vs-Rest (OvR) strategy, which converts the multi-label classification problem into multiple independent binary classification problems — one for each label.
The classifiers used include**:**

© 2025 IJMRSET | Volume 8, Issue 5, May 2025| DOI:10.15680/IJMRSET.2025.0805213

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Logistic Regression (LR): It is a fundamental machine learning algorithm employed for binary classification tasks. It is particularly effective in situations where the goal is to predict a categorical outcome with two possible classes. In the context of toxic comment classification, LR can be used to classify text as either "toxic" or "non-toxic" based on the features extracted from the text. Logistic Regression works by fitting a model to the data that predicts the probability of a comment belonging to the "toxic" class (1) or the "non-toxic" class (0).

$$\text{sigmoid}(z) = \frac{1}{1 + e^{-z}}$$

where z = wx +b

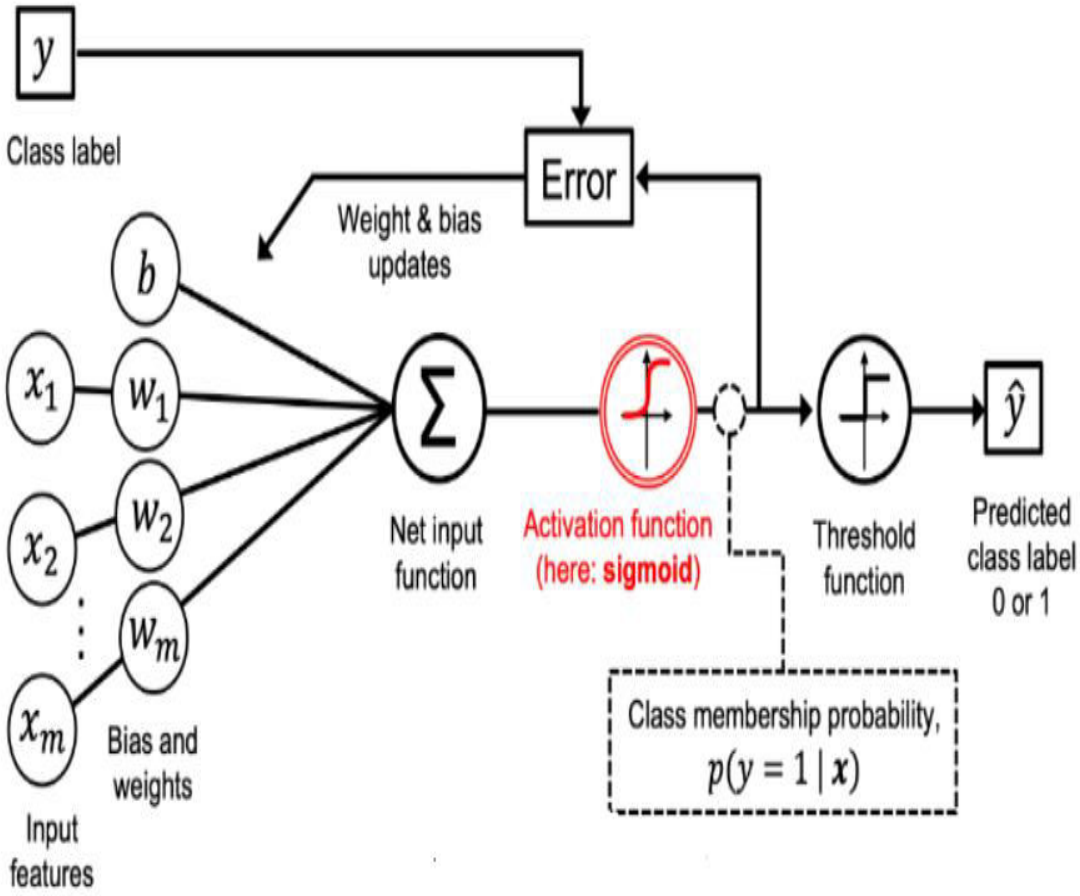


Fig. 3: Block Diagram for Logistic Regression

Naive Bayes: The Bayes theorem is a mathematical procedure for estimating conditional probabilities. A probability, as you may know, is the possibility of an event occurring. We call an event's probability if it has a chance of occurring. The Bayes theorem is the foundation of the Naive Bayes algorithm. It's primarily utilized for categorization tasks. In classification, we teach the model what each class belongs to using a labelled dataset, and then the model learns and classifies or labels a new dataset that has never been seen before. All of the features or variables in the Naive Bayes model are treated as independent of one another.

$$P(A|B) = \frac{(P(B|A) \cdot P(A))}{P(B)}$$

© 2025 IJMRSET | Volume 8, Issue 5, May 2025| DOI:10.15680/IJMRSET.2025.0805213

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

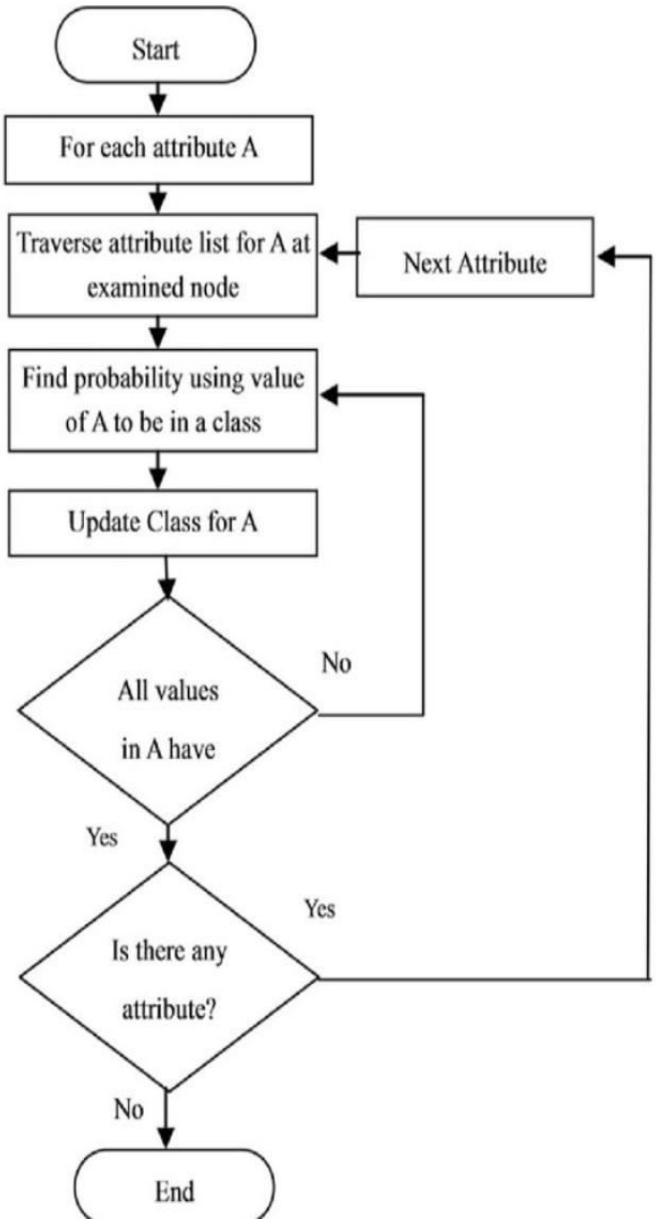


Fig. 4: Flow Chart for Naïve Bayes

Random Forest: Random Forest is a controlled machine learning approach to classification and regression problems. Uses an explicit majority for classification and a moderate delay to create a decision tree of different data. One of the main features of the random forest algorithm is the ability to manage a set of data with classified and continuous variables, such as regression and classification. As for the rating, it performs better than its competitors.

$$y^{\wedge}=\text{mode}(h1(x), h2(x), h3(x)\ldots\ldots, hn(x))$$

Where: n is Number of trees in the forest
y^ is Final aggregated prediction

© 2025 IJMRSET | Volume 8, Issue 5, May 2025| DOI:10.15680/IJMRSET.2025.0805213

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

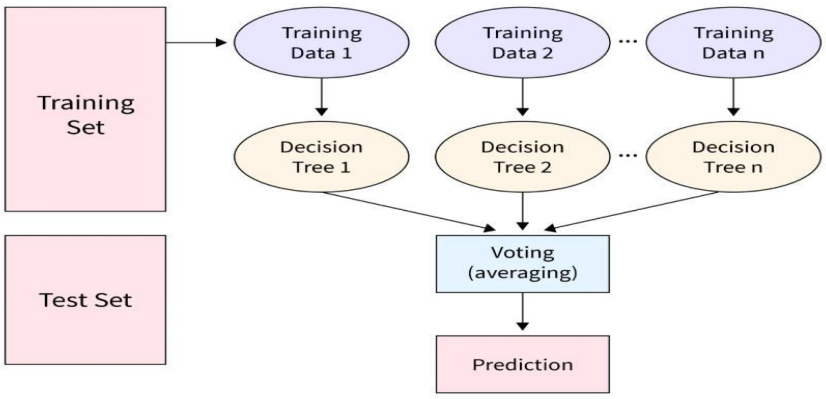


Fig. 5: Flow Chart for Random Forest

SVM: The support vector machine (SVM) is a machine learning technique with classification and regression monitoring. SVM defines a cloud page that classifies the boundaries between two data sets. SVM is often used to classify data, although it can also be used for regression. This is a fast and reliable method that works well with low data. Another advantage of SVM is that it can explore various input functions without increasing system problems, using different types of core functions.

$$F(x) = wx + b$$

where w is the weight vector, x is the feature vector, and b is the bias.

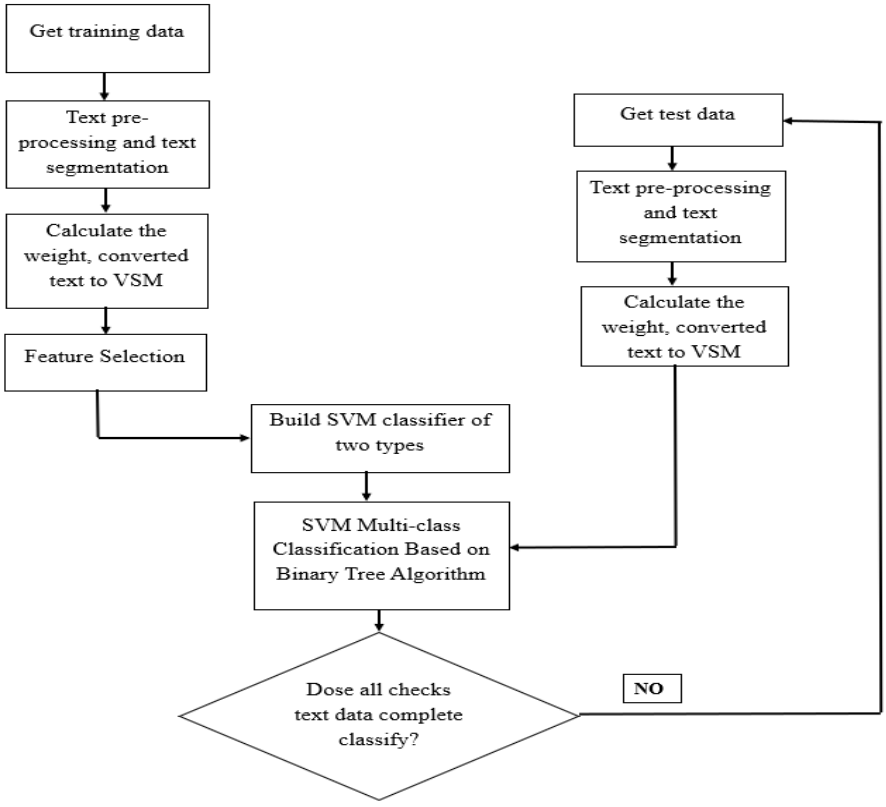


Fig. 6: Flow Chart for SVM

© 2025 IJMRSET | Volume 8, Issue 5, May 2025| DOI:10.15680/IJMRSET.2025.0805213

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Decision Tree: A decision tree is a simple classifier in the form of a hierarchical tree structure, it creates an upside-down tree to make predictions, starting at the top with a question about an important feature in your data, then branches out based on the answers. As you follow these branches down, each stop asks another question, narrowing down the possibilities. This question-and-answer game continues until you reach the bottom a leaf node where you get your final prediction or classification. The Decision Tree classifier operates by recursively splitting the data based on the most informative features. Uses Gini Impurity or Entropy to select the best word (feature) to split.

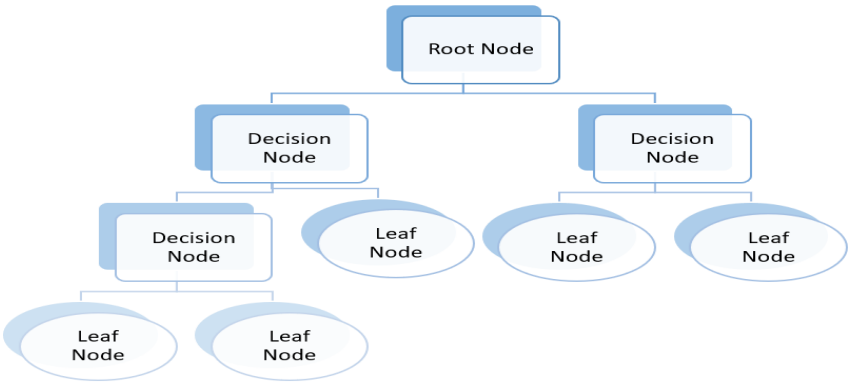


Fig. 7: Flow Chart for Decision Tree

KNN: A K Nearest Neighbor classifier is a machine learning model that makes predictions based on the majority class of the K nearest data points in the feature space. The KNN algorithm assumes that similar things exist in close proximity. KNN algorithm requires the data to be scaled first. Convert categorical columns into 0 & 1 so that no single feature dominates the distance metric. The KNN classifier operates by finding the K nearest neighbors to a new data point and then voting on the most common class among these neighbors. The model calculates the Euclidean Distance between a new datapoint and existing point.

$$d(p, q) = \sqrt{\sum_{i=1}^{n}(qi - pi)^2}$$

Where: d (p, q) =Distance between two points.
qi, pi=feature values of points.

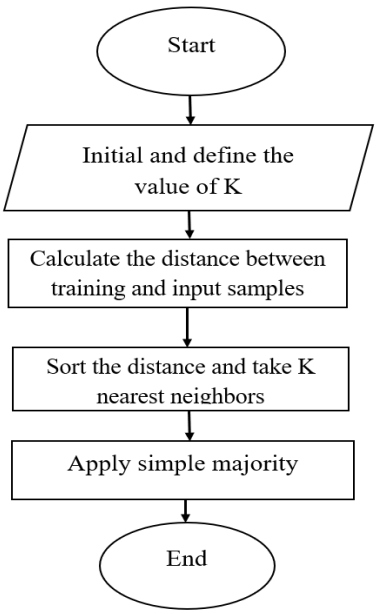


Fig. 8: Flow Chart for KNN

© 2025 IJMRSET | Volume 8, Issue 5, May 2025| DOI:10.15680/IJMRSET.2025.0805213

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

D. Model Evaluation

Evaluation metrics were computed on the test set using:

1. Accuracy: Overall correctness of the model.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

2. Precision: Precision evaluates the proportion of true positive predictions among all positive predictions made by the model. High precision indicates that the model produces fewer false positives.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

3. Recall: True positives among all actual positives.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

4. F1 Score: Harmonic mean of precision and recall.

$$\text{F1 Score} = 2\text{x}\frac{\text{Precision x Recall}}{\text{Precision} + \text{Recall}}$$

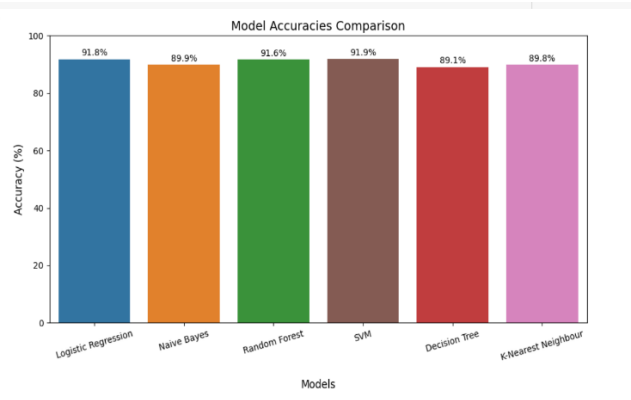
IV. RESULTS

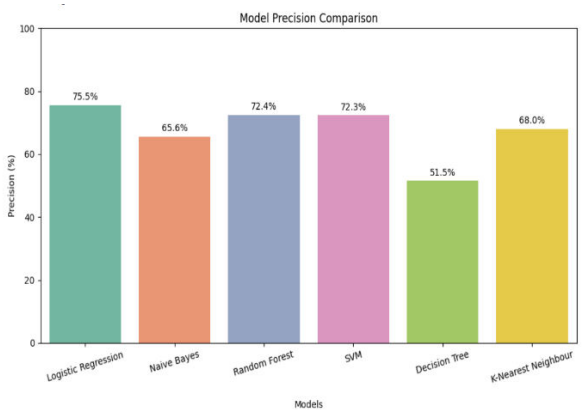
Each of the six machine learning models was trained and tested on the same dataset using 80:20 train-test split. The performance metrics are summarized in Table 1.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
SVM	91.9	72.3	46.4	55.4
Logistic Regression	91.8	75.5	39.1	49.6
Random Forest	91.6	72.4	31.4	41.0
Naïve Bayes	89.9	65.6	6.1	10.8
KNN	89.8	68.0	15.0	24.0
Decision Tree	89.1	51.5	47.3	48.8

Table 1

SVM emerged as the most balanced and high-performing model, making it suitable for deployment in moderation systems.



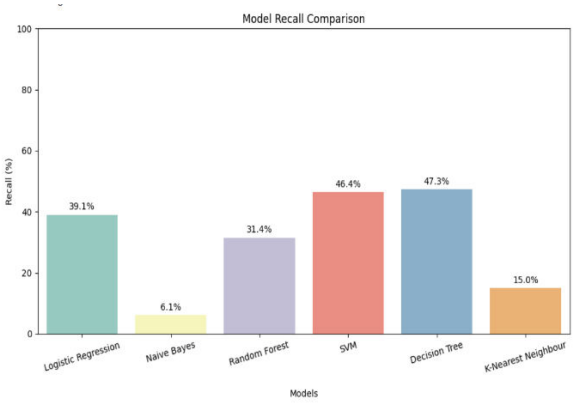


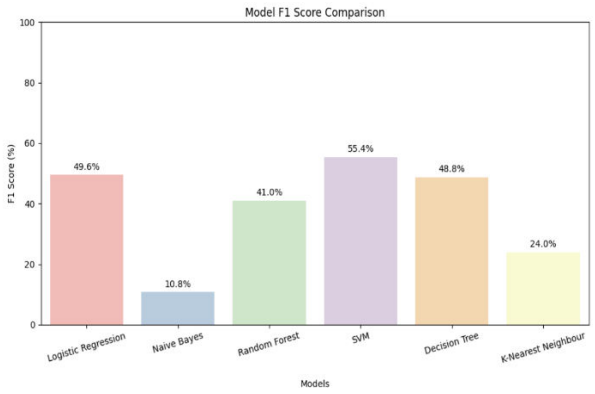
© 2025 IJMRSET | Volume 8, Issue 5, May 2025| DOI:10.15680/IJMRSET.2025.0805213

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

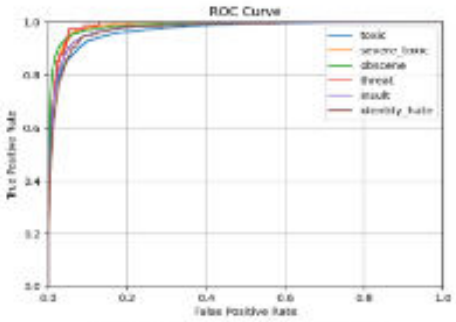
(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



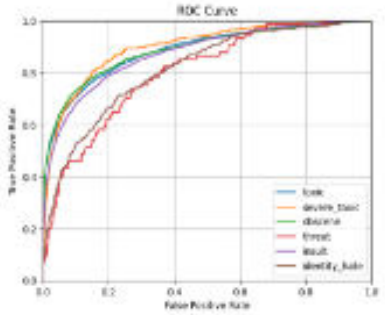


According to accuracy, we can conclude that the best model would be SVM since it had an accuracy of 91.9%. According to precision, we can conclude that the best model would be Logistic regression since it had a precision of 91.9%. According to recall we can conclude that the best model would be Decision Tree since it had a recall of 47.3%. According to F1 score we can conclude that the best model would be SVM since it had a F1 score of 55.4%.

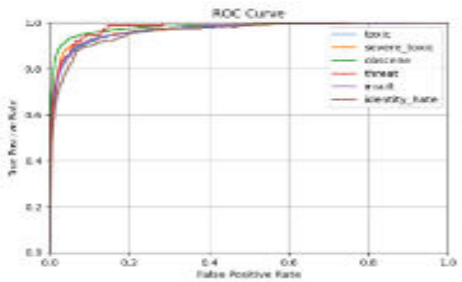
ROC Curves:



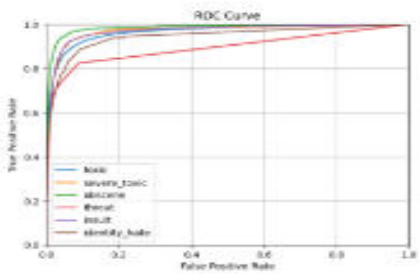
ROC for Logistic Regression



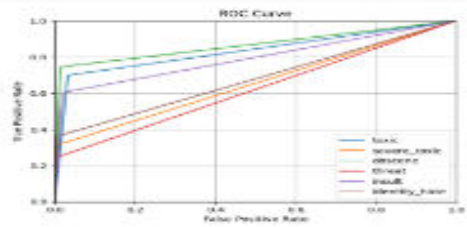
ROC for Naïve Bayes



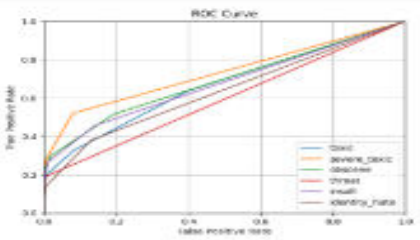
ROC for SVM



ROC for Random Forest



ROC for Decision Tree



ROC for KNN

V. CONCLUSION AND FUTURE WORK

This study presents a comprehensive machine learning framework for detecting and classifying toxic online comments using NLP and multi-label classification strategies. Among various classifiers, Support Vector Machine (SVM) proved most effective, achieving an F1 score of 55.4%. The study successfully demonstrates how traditional ML methods, coupled with thoughtful preprocessing and vectorization, can achieve strong results even in complex text classification scenarios.

VI.FUTURE WORK

Incorporate deep learning models like BERT or XLNet for contextual feature extraction.

- Address data imbalance using techniques such as SMOTE or cost-sensitive learning.
- Deploy the model as a real-time API for comment moderation systems.
- The evaluation metrics validate the model's ability to generate relevant and comprehensible captions for images, though they also highlight areas where linguistic richness and diversity could be improved.

REFERENCES

1. Abhishek Aggarwal and Atul Tiwari," Multi Label Toxic Comment Classification using Machine Learning Algorithms", International Journal of Recent Technology and Engineering (lJRTE), vol. 10, no. I, May 2021, ISSN 2277-3878.
2. P. A. Ozoh, A. A. Adigun and M. O. Olayiwola," Identification and Classification of Toxic Comments on Social Media using Machine Learning Techniques", International Journal of Research and Innovation in Applied Science (lJRIAS), vol. IV, no. XI, November 2019, ISSN 2454-6194.
3. [Liu, B., & Zhang, L. (2012). A Survey of Opinion Mining and Sentiment Analysis. Mining Text Data, 1(2), 87-132. Hu, M., Liu, B., & Zhang, L. (2013). Mining and Summarizing Customer Reviews. ACM Transactions on Knowledge Discovery from Data (TKDD), 7(2), 1-24.
4. Tsoumakas, G., & Katakis, I. (2007). Multi-label Classification: An Overview. Proceedings of the International Conference on Data Mining (ICDM), 22-25. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Le, Q., & Salakhutdinov, R. (2018). XLNet: Generalized Autoregressive Pretraining for Language Understanding. Advances in Neural Information Processing Systems (NeurIPS), 31.
5. Smith, J., & Johnson, A. (2018). Natural Language Processing Techniques. Springer. Brown, C., & Miller, R. (2019). Machine Learning for Multilabel Classification. Cambridge University Press.
6. 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.
7. V. Mishra and M. Tripathi," Detecting Toxic Comments Using Convolutional Neural Network Approach", Proceedings of the 2022 IEEE Conference on Intelligent Communications and Networks (CICN), 2022.
8. N. Kumar Singh and S. Chand," Machine Learning-based Multilabel Toxic Comment Classification", Proceedings of the International Conference on Computing Communication and Intelligent Systems (ICCCIS), 2022.
9. D. Abdul, Md. Shamimur and J. Rabbi," Detecting Abusive Comments in Discussion Threads Using Naive Bayes", Proceedings of the International Conference on Innovation in Software Engineering and Technology (ICISET), 2018.
10. R. Hooda, H. Kaijla, J. Hooda and G. Saini," Classification of Online Toxic Comments Using Machine Learning Algorithms", Proceedings of the International Conference on Innovative Computing and Communication Systems (ICICCS), 2020.
11. L. Jiahong, X. Yang, W. Weijia and Y. Jiaxin," Research on online user comments in artificial intelligence times", Proceedings of the International Conference on Artificial Intelligence and Cloud Computing (ITAIC), 2019.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH

IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | ijmrset@gmail.com |

www.ijmrset.com