Evaluating the Effectiveness of Generative and Discriminative Models in Detecting Toxic Comment

Student Id: 231618

Objective : Develop Models for Toxic Comment Classification

Why?

- Enhance online safety by automatically identifying and flagging toxic comments.
- a healthier and more respectful online environment by avoiding impact of abusive behaviour.
- Enable social media platforms to uphold community guidelines and policies effectively.
- Improve user experience by reducing exposure to harmful or offensive content.

Dataset Structure

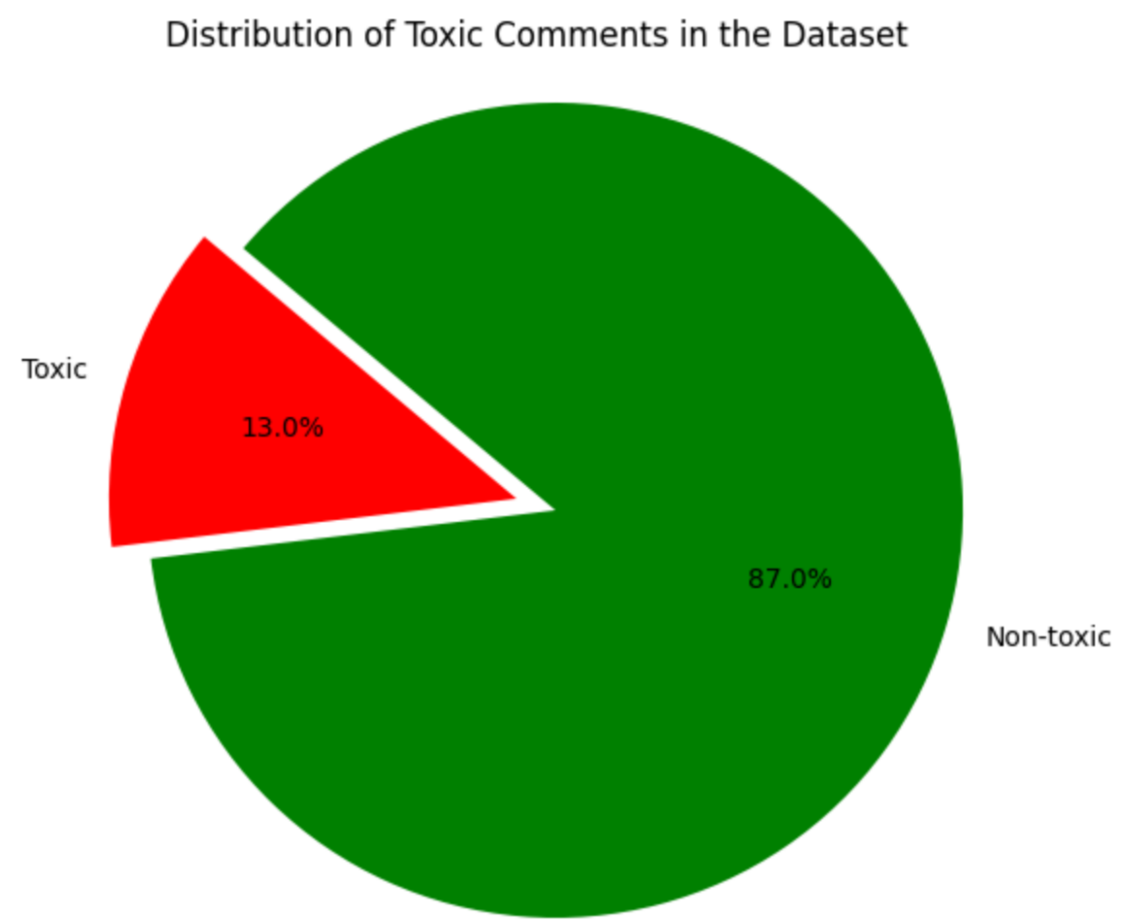
The dataset consists of three CSV files: Train, Test, and Valid. Each row contains the Following columns:

Comment Id

Comment

Split

Toxicity



Dataset Preprocessing

Why ?

Dataset pre-processing is essential for cleaning and enhancing data quality, extracting meaningful features, and ensuring consistent scales.

Steps:

- Performing Lowercase
- Removal of Special Character
- Removal of Stop Words
- Tokenization
- Lemmatization
- Stemming

Generative Approach:

Gaussian Naïve Bayes Classifier

- Gaussian Naive Bayes (GNB) is a probabilistic-based classifier commonly used for text classification tasks.
- It assumes that features follow a normal distribution and are independent of each other.
- The class with the highest probability is selected as the prediction.
- In text classification, GNB calculates each word occurring in toxic and non-toxic comments.
- It combines these probabilities using Bayes' theorem to determine the overall probability of the comment being toxic or non-toxic.
- Despite its simplistic assumptions, GNB performs well in practice, especially with continuous data.

Discriminative Approach:

Gradient Boost Classifier

- Gradient Boosting Classifier (GBC) is a powerful machine learning algorithm commonly used for text classification tasks.
- It builds an ensemble of decision trees sequentially, with each tree focusing on correcting the errors of its predecessor.
- GBC calculates the probability of a given comment belonging to each class and combines these probabilities using a weighted sum.
- In text classification, GBC identifies the most informative features to distinguish between toxic and non-toxic comments, resulting in a highly accurate and robust model.
- Despite its complexity, GBC is efficient and provides exceptional performance in handling the challenges of text data.

Why this Models?

- Among the implemented models, Gradient Boosting Classifier (GBC) and Gaussian Naive Bayes (GNB) showed good performance, possibly due to the limited dataset favoring simpler models.
- GNB's simplicity and assumption of feature independence contributed to its effectiveness in predicting the "out label" model, despite limited data.
- GBC demonstrated strong discriminative ability, outperforming models like Support Vector Classifier (SVC) and Long Short-Term Memory (LSTM), making it suitable for toxicity detection tasks.
- The decision to use both GBC and GNB was justified by their respective strengths: GBC's robustness in handling complex datasets and GNB's effectiveness with simplistic assumptions, especially in tasks with limited data.

Generative Model Implementation

- Data pre-processing ensured uniformity and readiness for analysis.
- TF-IDF (Term Frequency-Inverse Document Frequency) transformed text data into numerical features.
- Gaussian Naive Bayes classifier was trained to predict comment toxicity based on TF-IDF features.
- Model performance was evaluated on the validation set, and metrics were calculated to show effectiveness.

Discriminative Model Implementation

- Similar pre-processing steps were applied to ensure consistency in the training and validation datasets.
- TF-IDF was used to convert text data into numerical features.
- Gradient Boosting Classifier was trained through a pipeline approach to predict comment toxicity.
- Model performance was evaluated on the validation set to assess effectiveness.

Performance result

Metric	Discriminative Model	Generative Model
Accuracy	0.844	0.595
Recall	0.055	0.351
Precision	0.186	0.126
F1 Score	0.084	0.185
False Positives	Fewer	More
False Negatives	More	Fewer

Accuracy Metric:

•Discriminative model (Accuracy: 0.844) significantly outperforms the generative model (Accuracy: 0.595).

•Generative model captures a substantial portion of toxic comments despite lower accuracy, indicating effectiveness in identifying such instances.

Performance Result

Recall Metric:

- Generative model exhibits higher recall (0.351) compared to discriminative model (0.055).
- Higher recall for generative model may lead to higher false positive rate.

Precision Metric:

- Discriminative model shows slightly better precision (0.186) compared to generative model (0.126), indicating fewer false positives.

F1 Score Metric:

- Generative model has slightly higher F1 score (0.185) compared to discriminative model (0.084), indicating better balance between precision and recall.

Confusion Matrix Metric:

- Discriminative model has significantly fewer false positives compared to generative model, leading to higher precision.

SoTA Comparison

Model Performance:

- SoTA ‘s Model Bidirectional LSTM outperforms compared to GNB and GBC models with an F1-score of 0.94.

Hyperparameter Tuning:

- Extensive tuning included parameters like layers, optimizer, activation function, etc., enhancing model effectiveness.

Optimization Techniques:

- Adam optimizer, batch size of 32, binary cross-entropy loss, sigmoid activation used for refinement.

Dataset Characteristics:

- Dataset richness enabled effective learning of toxic comment patterns, contributing to SoTA model's superiority.

Implications and Applications:

- SoTA model's success highlights advanced NLP's importance in toxic comment detection, applicable in content moderation, social media monitoring, etc.

Example

Comment ID	GT	Generative	Discriminative
257542852	1	Predicted	Predicted
78309297	1	Predicted	Predicted
253911271	1	Predicted	Predicted
419746843	1	Not Predicted	Not Predicted
635744705	1	Predicted	Not Predicted

Example

Comment 257542852 and Comment 78309297:

- Both comments contain explicit language and are likely considered toxic.
- Both models correctly predicted them as toxic, showcasing their effectiveness in identifying such instances.

Comment 253911271:

- This comment seems to be a factual statement without any explicit language or toxic content.
- However, both models incorrectly predicted it as toxic, indicating a misclassification issue.

Comment 419746843:

- This comment discusses the manipulation of a report and uses offensive language.
- The discriminative model correctly predicted it as toxic, while the generative model failed to do so.
- This highlights the discriminative model's ability to capture toxicity.

Comment 635744705:

- This comment appears to be an advertisement or a statement about copyright issues.
- While it contains offensive language, it may not necessarily be classified as toxic in the context of the dataset.
- The generative model incorrectly predicted it as toxic, while the discriminative model correctly identified it as non-toxic.
- This emphasizes the importance of context and the challenges in classifying comments accurately.

Limitation

Dataset Imbalance:

- The dataset used for training and evaluation exhibits significant class imbalance, with fewer toxic comments compared to non-toxic comments.
- Class imbalance poses challenges for both models, as they may struggle to effectively learn patterns associated with toxic comments.
- Imbalanced datasets often result in biased models that favor the majority class, leading to lower performance metrics for the minority class.
- Both models demonstrate relatively good accuracy and precision but struggle with low recall scores, indicating difficulty in identifying toxic comments effectively.

Future Enhancement

Dataset Improvement:

- Utilize a more extensive dataset to cover a broader range of toxic comments, addressing biases and improving model generalization.

Experimenting high level Model Architecture:

- Explore advanced deep learning architectures like Convolutional Neural Networks (CNN) or Transformer-based models to capture complex patterns within the data.

Text Pre-processing:

- Apply advanced text pre-processing techniques such as character-level tokenization to enhance feature extraction and model performance.

Ensemble Methods:

- Investigate ensemble methods like model blending or stacking to combine the strengths of different models and improve overall predictive performance.

Hyperparameter Optimization:

- Optimize the hyperparameters of existing models to fine-tune their performance and achieve better results.

Addressing Class Imbalance:

- Handle class imbalance issues through techniques like oversampling or using appropriate performance metrics that prioritize recall, focusing on effectively identifying toxic comments.

Knowledge Gained

Proficiency in NLP:

- Gained proficiency in implementing models for text data analysis and understanding NLP fundamentals.

Importance of Hyperparameters:

- Identified the significance of hyperparameters and their role in improving model performance.

Understanding Evaluation Metrics:

- Learned the importance of evaluation metrics in assessing model effectiveness.

Model Comparison:

- Compared and contrasted various models (e.g., Gaussian Naive Bayes vs. Gradient Boosting Classifier) for identifying toxic comments.

Continuous Experimentation:

- Highlighted the necessity of continuous experimentation and exploration of different techniques in model development.

Future Directions:

- Equipped to handle challenges such as dataset enhancements, exploring advanced model architectures, and addressing class imbalance.

Conclusion:

- Completion of the project provided valuable insights into enhancing text classification models' performance and paved the way for future improvements in this domain.

Thank You !!!!!!!