Detecting Bengali Hate Speech: A Machine Learning and Transformer-Based Approach

Sanjana Akhter
Department of CSE
20210104150
Dhaka, Bangladesh
sanjana.cse.20210104150@aust.edu

Md. Mostafizur Rahman
Department of CSE
20210104151
Dhaka, Bangladesh
mostafizur.cse.20210104151@aust.edu

Mahajabin Haque
Department of CSE
20210104154
Dhaka, Bangladesh
mahajabin.cse.20210104154@aust.edu

***Abstract*—With the rise of social media, the prevalence of hate speech on online platforms has increased significantly, leading to harmful impacts on society. Hate speech is a global issue, including in Bangladesh. While there has been considerable research on hate speech detection globally, studies on hate speech in the Bengali language remain limited due to the scarcity of high-quality datasets. In this study, we apply machine learning algorithms such as Gradient Boosting, XGBoost, SVM, Naive Bayes, and KNN, as well as transformer-based models including various versions of the pretrained BERT model, to detect Bengali hate speech in social media text from a dataset containing 30,000 entries. This work is carried out using Python in a Google Colab environment. Our research shows that language-specific transformer models like BanglaBERT perform best on the dataset, demonstrating their capability in detecting hate speech in Bengali.**

***Index Terms*—Bengali hate speech detection, NLP, Machine Learning, SVM, XGBoost, Gradient Boosting, Naive Bayes, KNN, BanglaBERT, mBERT, XLM-Roberta, Text classification.**

I. Introduction

With more people using social media, the rising issue of hate speech has become a major concern. It spreads negativity and affects individuals and communities. There has been a lot of research regarding hate speech globally and in many languages. There has been research on Bengali hate speech as well, but the biggest issue is finding a dataset that is rich enough for the task.

Regional language hate speech tasks are different from other NLP tasks because the type of words used in hate speech datasets is different from others.

In this research, we focus on detecting Bengali hate speech using machine learning and transformer-based models. We use different algorithms, including Gradient Boosting, XGBoost, SVM, Naive Bayes, KNN, and transformer-based models like BanglaBERT, mBERT, and XLM-RoBERTa. Our dataset contains 30,000 Bengali comments collected from social media. We preprocessed the dataset according to our needs to ensure better model performance. The models are evaluated and compared.

The rest of the paper is structured as follows: Section II is a literature review on hate speech detection for the Bengali language. Section III describes the dataset, the preprocessing steps, and data visualization. Section IV includes the methodology, including machine learning and transformer-based models. Section V discusses the results and compares the performance of different models. Finally, Section VI is the conclusion.

II. Literature Review

M. Jobair et al. (2024) [1] conducted a comparative study on Bengali hate speech detection using deep learning models, including CNN, LSTM, Bi-LSTM, GRU, and BERT. The study introduced an 8,600-comment dataset, categorized into five themes. Among the models, BERT achieved the highest accuracy (97%) on an existing dataset of 30,000 records, surpassing LSTM and Bi-LSTM while it had 80% accuracy on the new dataset.

Md. R. Karim et al. (2023) [2] explored multimodal hate speech detection using textual and image-based models, incorporating Bi-LSTM, Conv-LSTM, Bangla BERT, mBERT, and XLM-RoBERTa. The study reported that XLM-RoBERTa outperformed other models, achieving the highest F1 score of 82%.

F. Haider et al. (2024) [3] introduced a multi-label hate speech detection dataset for transliterated Bengali (Banglish), evaluating models such as BanglaBERT, BanglishBERT, and TB-mBERT. TB-mBERT achieved the highest binary classification accuracy (82.57%), while BanglaBERT excelled in multi-label classification (54.08%).

S. Akter et al. (2024) [4] proposed a hybrid deep learning model integrating BERT with BiLSTM and SVM classifiers. The best-performing model which is a BERT-CNN hybrid, achieved 94.44% accuracy and it outperformed traditional deep learning architectures like LSTM and GRU.

M. Islam et al. (2022) [5] investigated traditional ML models such as Logistic Regression, Naïve Bayes, Random Forest, SVM, and KNN for Bengali hate speech detection.

BanglaBERT outperformed these models, achieving 93% accuracy, whereas traditional classifiers like Random Forest and Naïve Bayes lagged behind at 67% and 65%, respectively.

M. Das et al. (2023) [6] focused on transformer-based models, including mBERT, XLM-RoBERTa, IndicBERT, and MuRIL, for detecting hate speech in Bengali and Romanized Bengali. XLM-RoBERTa achieved the highest accuracy (83.3%) for Bengali, while MuRIL showed competitive performance, emphasizing the role of multilingual models.

III. Data Analysis

The dataset used in our research contains 30,000 sentences, with each entry labeled as either "hate" or "no hate" along with a category for them. The dataset is imbalanced because 20,000 instances are labeled as "no hate" and the rest are labeled as "hate." To prepare the data for model training and ensure balanced representation, we perform several preprocessing steps. Figure 1 shows the distribution of the "hate" and "no hate" categories in the dataset.

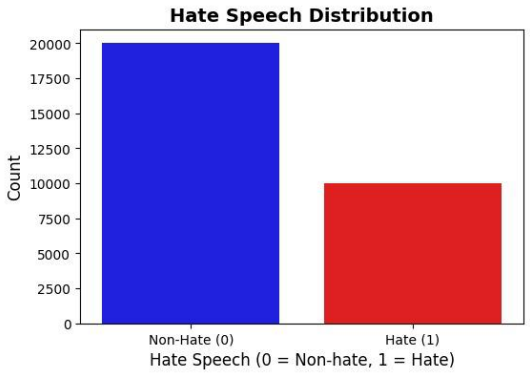

Fig. 1. Overview of the dataset structure

Initially, we clean the text by removing unwanted characters, symbols, and extra spaces. There were also some English words and numbers in the sentences, and we removed them to keep only the Bengali words. The sentences are then tokenized with the help of the Indic NLP library, which breaks the text into individual words suitable for analysis. We further clean the data by removing Bengali stopwords using the NLTK stopwords list. Now, it is ready for machine learning tasks. As we used TF-IDF for machine learning models, and it is required in TF-IDF to remove stopwords. However, for transformer-based models, stopwords are necessary to understand the context, so we didn't remove the stopwords for BERT models; rather, we used the cleaned sentence part. After cleaning and tokenization, we shuffle the data because, in our original dataset, the first 10,000 entries were labeled as "hate," and the remaining 20,000 were labeled as "no hate." The shuffling step ensured that the data is evenly distributed across both categories.

Following preprocessing, the tokenized text is ready for model input. For transformer-based models like BanglaBERT, the sentences are encoded using the pre-trained BanglaBERT tokenizer, with padding and truncation applied to ensure consistent input length.

After preprocessing, the dataset is balanced, cleaned, and tokenized, making it suitable for model training.

IV. Methodology

A. Overview

After preprocessing as discussed in the data analysis section we selected a combination of machine learning and transformer-based models to classify Bengali hate speech. The models were trained, tuned, and validated on a preprocessed dataset.

To improve the classification performance we first experimented with traditional machine learning models, which showed moderate accuracy. However, to get better accuracy we further implemented transformer-based models.

B. Machine Learning Algorithms

For initial experimentation, we used Support Vector Machine (SVM), Naïve Bayes, Gradient Boosting, K-Nearest Neighbors (KNN), and XGBoost classifiers.

These models function based on statistical feature extraction techniques. SVM finds an optimal hyperplane for classification, while Naïve Bayes works on the assumption of conditional independence. Random Forest and Gradient Boosting use ensemble learning to improve predictive performance, whereas XGBoost enhances gradient boosting through parallelization. KNN operates on the principle of distance-based similarity.

These models were chosen due to their efficiency in text classification tasks. However, they struggled to capture the semantic relationships in Bengali text, leading to limited performance, necessitating the shift to transformer-based models.

C. Transformer-Based Models

Since traditional Machine learning algorithms were unable to perform well on the dataset we then applied transformer based model to get higher performance.

1) **BanglaBERT:** A language-specific model trained on large-scale Bengali text corpora. This was chosen as it is fine-tuned specifically for Bengali, making it more suitable for understanding language-specific nuances.
2) **Multilingual BERT (mBERT):** Both cased and uncased versions of mBERT were used to evaluate their effectiveness. mBERT is trained on 104 languages and allows for multilingual text understanding, but the cased version retains capitalizations, whereas the uncased version normalizes case sensitivity.
3) **XLM-RoBERTa (XLM-R):** A robust multilingual model trained on a diverse corpus, including Bengali. It is optimized for cross-lingual understanding, making

it a strong candidate for handling code-mixed Bengali text.

D. Hyperparameter Tuning

To optimize model performance, we tuned several hyperparameters for both machine learning and transformer-based models.

1) **Machine Learning Models**
   a) **SVM:**
      - Kernel = linear: Suitable for high-dimensional text data. Different Kernel trick was used while linear one performed best.
      - C = 1: Controls margin width; a balanced value prevents overfitting. It is the regularization parameter.
      - Class Weight = balanced: Handles class imbalance.
   b) **XGBoost:**
      - n_estimators = 700: Ensures sufficient boosting rounds for feature learning.
      - learning_rate = 0.1: Balances convergence speed and performance.
      - max_depth = 8: Prevents overfitting while capturing complex patterns. Depth 5-8 was used while 8 performed best.
      - scale_pos_weight: Adjusted for class imbalance to prioritize underrepresented classes.
   c) **Gradient Boosting:**
      - n_estimators = 500: Optimized based on dataset size to prevent overfitting.
      - learning_rate = 0.1: Ensures gradual weight updates.
      - max_depth = 6: Prevents excessive complexity.
      - subsample = 0.8: Helps reduce variance and overfitting.
   d) **Naïve Bayes:**
      - alpha = 0.1: Smoothing parameter to handle zero probabilities.
   e) **KNN:**
      - n_neighbors = 5: Balances between overfitting (low k) and underfitting (high k). Also tested with k=7 and k=9.
      - metric = Minkowski, p = 2: Uses Euclidean distance for similarity measurement.
2) **Transformer-Based Models (BanglaBERT, mBERT, XLM-R)**
   a) Batch Size = 16: Balances GPU memory usage and training efficiency. Batch size 32 needs more GPU memory but didn't really increase performance that much.
   b) Learning Rate = 2e-5: Small value ensures stable fine-tuning.
   c) Weight Decay = 0.01: Prevents excessive weight updates, improving generalization.
   d) Optimizer = AdamW: Preferred for transformer models to improve convergence.
   e) Loss Function = Weighted Cross-Entropy: Addresses class imbalance effectively.
   f) Training Epochs = 5: Ensures convergence without overfitting. Also tested with 7 epochs but the result is almost the same.
   g) Evaluation Strategy = Epoch-wise validation: Assesses performance after every epoch.

E. Model Evaluation

After training, we evaluated models using multiple performance metrics.

Accuracy: Measures the overall correctness of predictions, calculated as the ratio of correctly classified instances to the total number of instances.

Precision: Represents the proportion of correctly predicted positive cases out of all predicted positive cases. High precision indicates fewer false positives, making it crucial when minimizing incorrect hate speech flags.

Recall: Measures the proportion of actual positive cases that were correctly identified. A higher recall value means fewer false negatives, which is important for capturing all instances of hate speech.

F1-Score: The harmonic mean of Precision and Recall, providing a balanced metric when dealing with class imbalance. A high F1-score indicates that both false positives and false negatives are minimized.

Confusion Matrix: A visualization of the model's predictions versus actual values. It highlights the distribution of true positives, true negatives, false positives, and false negatives, offering insight into misclassification patterns.

V. Result Analysis

At first we applied traditional machine learning algorithms such as XGBoost, Naïve Bayes, Gradient Boosting, SVM, and KNN. These models struggled to achieve high accuracy, with the best-performing model, XGBoost, achieving only 74.32% accuracy, while KNN performed the worst at 66.98%. The poor performance of ML models can be attributed to dataset imbalance. Even after applying weighted values and shuffling, the models still failed to improve significantly.

Model	Accuracy	Precision	Recall	F1-score
XGBoost	74.32%	74.22%	74.15%	71.23%
Naive Bayes	73.98%	72.98%	74.11%	71.34%
Gradient Boosting	73.93%	73.12%	74.05%	72.31%
SVM	70.9%	71.43%	71.29%	71.21%
KNN	66.98%	65.33%	67.21%	66.12%

TABLE I
MACHINE LEARNING MODEL PERFORMANCE COMPARISON

To enhance classification performance, we applied transformer-based models, which have demonstrated superior results in NLP tasks. Among them, Bangla BERT outperformed all other models, achieving 90.03% accuracy. This is because Bangla BERT is pre-trained specifically on Bengali text, making it highly effective in understanding Bengali linguistic patterns.

Model	Accuracy	Precision	Recall	F1-score
Bangla BERT	90.03%	90.04%	90.03%	90.04%
mBERT Uncased	89.78%	89.74%	89.78%	80.76%
mBERT Cased	89.55%	89.53%	89.55%	89.54%
XLM-R	88.38%	88.67%	88.38%	88.48%

TABLE II
TRANSFORMER BASED MODELS PERFORMANCE COMPARISON

We also tested multilingual BERT models, where mBERT Uncased (89.78%) outperformed mBERT Cased (89.55%). Since Bengali is not case-sensitive, the uncased version handled tokenization more effectively by avoiding unnecessary token splits. XLM-R achieved 88.38% accuracy, slightly lower than other transformer models, possibly due to its multilingual training, which makes it less optimized for Bengali than Bangla BERT.

VI. CONCLUSION AND FUTURE WORK

In this research we explored the effectiveness of machine learning and transformer-based models in detecting Bengali hate speech. We found that traditional machine learning models achieved lower accuracy due to their limitations and the dataset imbalance, whereas transformer-based models performed significantly better outperforming ML models.

Our findings show that transformer-based models are more effective in handling Bengali hate speech classification. While BanglaBERT performed the best, mBERT models also showed strong results, proving that although language-specific models are superior, multilingual models can still perform well. Our research highlights the importance of using language-specific models for NLP tasks.

For future work we can focus on expanding the dataset, improving model generalization, and exploring multimodal approaches, as discussed in our literature review. If possible we will try to combine transformer models with deep neural networks like LSTM and GRU. Additionally, optimizing model architectures and fine-tuning parameters could further enhance accuracy and efficiency.

REFERENCES

[1] Md. Jobair, D. Das, N. B. Islam, and M. Dhar,Bengali Hate Speech Detection with BERT and Deep Learning Models, Preprint, Aug. 2023. Available: https://www.researchgate.net/publication/372902782.

[2] Md. R. Karim, S. K. Dey, T. Islam, M. Shajalal, and B. R. Chakravarthi, Multimodal Hate Speech Detection from Bengali Memes and Texts,in Proc. Int. Conf. Multimodal Interaction, 2023, pp. 1-9. Available: https://arxiv.org/abs/2204.10196.

[3] F. Haider, F. T. Shifat, M. F. Ishmam, D. D. Barua, M. S. U. R. Sourove, M. Fahim, and M. F. Alam, BANTH: A Multi-label Hate Speech Detection Dataset for Transliterated Bangla, Preprint, 2024. Available: https://arxiv.org/abs/2410.13281.

[4] S. Akter, M. Hosen, H. K. Mehedi, A. H. Shihab, et al., BengaliHateCB: A Hybrid Deep Learning Model to Identify Bengali Hate Speech Detection from Online Platform, 2024. Available: https://www.researchgate.net/publication/380828242.

[5] M. Islam, M. S. Hossain, and N. Akhter, Hate Speech Detection Using Machine Learning In Bengali Languages, Preprint, 2022. Available: https://www.researchgate.net/publication/363027993.

[6] M. Das, S. Banerjee, P. Saha, and A. Mukherjee, Hate Speech and Offensive Language Detection in Bengali, in Proc. AACL, 2023. Available: https://aclanthology.org/2022.aacl-main.23/.

[7] M. Al-Ayyoub, W. Khulief, and M. N. Al-Kabi, Detection of Hate Speech using BERT and Hate Speech Word Embeddings, Appl. Artif. Intell., vol. 37, no. 1, pp. 1-20, 2023. DOI: 10.1080/08839514.2023.2166719.

[8] J. S. Malik, H. Qiao, G. Pang, and A. van den Hengel, Deep Learning for Hate Speech Detection: A Comparative Study, Int. J. Data Sci. Anal., vol. 15, pp. 65-77, 2023. Available: https://arxiv.org/abs/2202.09517.