https://doi.org/10.65091/icicset.v2i1.3

Hate Speech Detection in Nepali Social Media: A Comparative Analysis of Machine Learning and Transformer-based Approaches

Palisha Shakya
Department of Computer Engineering
Nepal College of Information Technology
Pokhara University, Nepal
palshakya2@gmail.com

Suraj Chand
Department of Computer Engineering
Nepal College of Information Technology
Pokhara University, Nepal
surazchand842@gmail.com

Lalit Buda Pal
Department of Computer Engineering
Nepal College of Information Technology
Pokhara University, Nepal
lalitpal091091@gmail.com

Manil Vaidhya*
Department of Computer Engineering
Nepal College of Information Technology
Pokhara University, Nepal
manil.baidhya@ncit.edu.np

*Corresponding author

***Abstract*—The growth of hate speech on social media presents serious problems for low-resource languages like Nepali, as automated detection systems for the Nepali language are underdeveloped. This paper presents a comprehensive approach for detecting hate speech in Nepali social media by combining lexicon-based feature engineering with machine learning and transformer-based models. The NEHATE dataset, which included 13,505 annotated tweets from Nepal's municipal election discourse in 2022, was used for this study along with a curated lexicon of 1,077 offensive terms divided into classes such as Politics (115 terms), Race (77), Vulgar (50), Disability (37), and Gender (15). The lexicon includes 159 taboo terms and 158 severely offensive terms, with ratings ranging from 1 (mild) to 5 (severe). The methodology uses the lexicon for feature engineering rather than training data, extracting offensive term counts, maximum offensiveness scores, taboo presence, and category-specific indicators. Two traditional machine learning models – Naive Bayes and Gradient Boosting were applied using character-level TF-IDF (n-grams 2–5) with lexical characteristics. Additionally, three multilingual transformers—mBERT, XLM-RoBERTa, and MuRIL were fine-tuned. The pre-processing pipeline handles both Devanagari script and Romanized Nepali text common on social media. Experimental results on an 80-20 train-test split demonstrate that Random Forest Classifier achieves the best performance among traditional machine learning models (F1-score: 0.847, AUC-ROC: 0.912), while MuRIL outperforms other transformer models (F1-score: 0.863). Also for the ablation study lexicon enhanced features improve F1-score over TF-IDF alone. This study demonstrates that lexicon-enhanced feature engineering significantly improves hate speech detection in low-resource languages and provides practical recommendations for developing content moderation systems for Nepali-speaking communities.**

***Index Terms*—Hate Speech Detection, Romanized Nepali, Natural Language Processing (NLP), Transformer Models, Multilingual BERT (mBERT), MURIL**

I. Introduction

The rapid advancements in digital technologies have significantly changed how people communicate. Traditional face-to-face interactions have been gradually replaced by online communications. This shift has made information exchange faster and more accessible but it has also contributed to the rise in harmful behaviors such as online vulgarity, harassment, and unethical language use. As more people in the country use smartphones and the internet, the Nepali-language content on social media platforms like Twitter, Facebook and Instagram is increasing. This also causes an increase in online hate speech. Hate speech detection in the Nepali language faces many difficul- ties as it is a low-resource language. These difficulties include scarcity of annotated datasets, the morphological complexity of the Nepali language, the frequency of code-mixing between Hindi, English, and Nepali, the usage of both Romanized and Devanagari scripts, and the lack of standardized pre-processing methods. Despite these challenges, large-scale content filtering requires automatic identification methods, since the amount of user-generated material makes it impossible to inspect manually. Through an empirical investigation, this research addresses the issue of hate speech identification in Nepali social media. Our contributions include a comparison of transformer-based and traditional machine learning techniques, a thorough analysis of existing Nepali hate speech datasets, the development of a lexicon-based feature extraction framework, and helpful recommendations for creating hate speech detection systems for low-resource languages.

II. Research Objectives

To develop an effective hate speech detection system for Nepali social media content by combining lexicon-based fea-

ture engineering with machine learning and transformer-based models.

III. Literature Review

Recent advancement in technologies has changed the way humans interact with each other. Online platforms have significantly replaced traditional face-to-face communication. This has raised concerns such as online vulgarity, harassment and unethical language. Detection of such content has been studied significantly in high resource languages like English but research on low resource languages like Nepali still faces some computational complexities.

Although several multilingual pretrained language models such as mBERT, XLM-RoBERTa, IndicBERT, and MuRIL claim promising cross-lingual performance, their true effectiveness on real-world Nepali text, especially romanized Nepali, has not been thoroughly evaluated. Sellars defines hate speech as verbal or written abuse that is directed towards a certain group of people, often because of their race, beliefs, or sexual orientation [1].

Deep learning (DL) models such as CNN , BERT, and MulRIL were used to benchmark a study in hate speech identification in Devanagari. With an F1 score of 0.72, MuRIL outperformed all other models [3]. MuRIL is pretrained on 17 Indian languages and their transliterations. XLM-RoBERTa, a large multilingual model, offers cross-lingual capabilities by training on diverse data from multiple languages.

IndicBERT focuses on 12 Indian languages, including Devanagari(Hindi and Marathi), and uses a lightweight structure ideal for efficient processing [3]. In a study performed , IndicBERT results were lower than anticipated, possibly because the dataset contained Nepali text, for which IndicBERT may not have been optimised [4]. The results of the combination of IndicBERT and LSTM CNN were found to be not satisfactory [4]. On the contrary, another study on devanagari script used different ML ,DL and transformer models such as Logistic Regression (LR), Support Vector Machine (SVM),Multinomial Naive Bayes (MNB), GradientBoosting Classifier (GBC),CNN, BiLSTM, CNN+BiLSTM, m-BERT, IndicBERT, MuRIL, and XLM-R. concluded that, IndicBERT and MuRIL outperformed ML and DL models by achieving a macro F1-score of 0.6785 among transformer based models. The XLM-R model also obtained a moderate result with a 0.6608 macro F1 Score. IndicBERT is the best model due to its higher precision value than MuRIL [5].

IV. Methodology

The proposed work uses lexicon based feature engineering, transformer based model and traditional machine learning model for detecting the hate speech from the dataset.

A. Data Collection

The primary dataset utilized is the NEHATE corpus, comprising 13,505 tweets from the local elections in Nepal held in 2022. These tweets are classified for hate speech, accompanied by additional labels for targets (Individual, Organization, Community) [2]. The other data set also contains a lexicon of offensive words, which includes 1,078 terms assigned offensiveness scores ranging from 1 to 5, alongside indicators for taboo and categories such as politics, race, gender, religion, and disability.

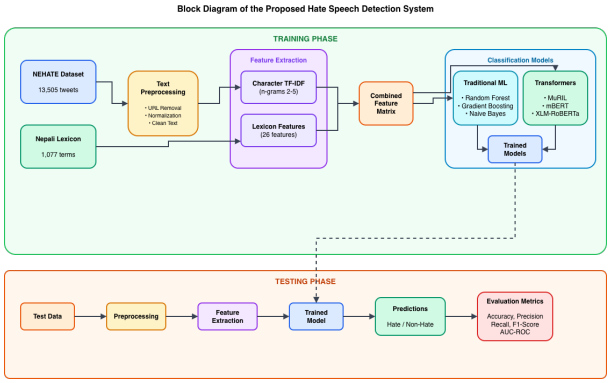

Fig. 1. Block Diagram of the proposed model

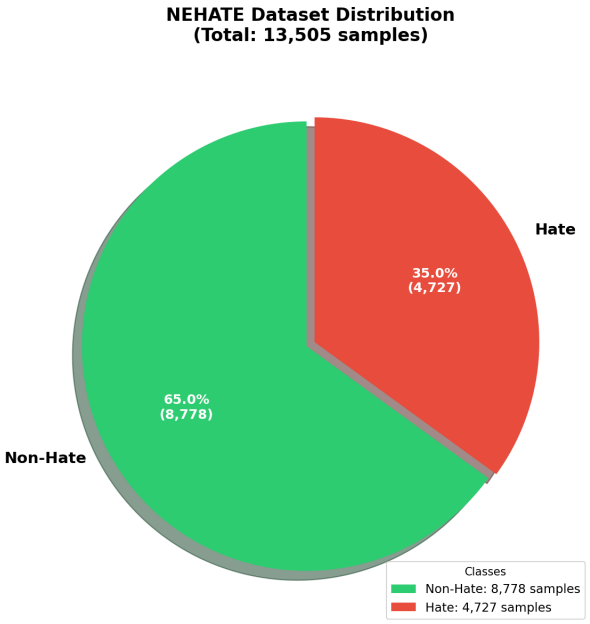

Fig. 2. Distribution of NEHATE dataset

B. Exploratory Data Analysis

Word Frequency analysis was done to understand the linguistic characteristics of hate and non-hate speech in Nepali. Figure 3 represent the word clouds comparing the most frequent terms in each class. The hate speech vocabulary is dominated by ethnic slurs (sala, muji, randi), targeted group references (madhesi, pahadi, muslim, christian), and violent action words (marnu parcha, hataunu parcha). In contrast, non-hate speech contains neutral political discourse terms (sarkar, chunab, niti) and everybody vocabulary (ramro, namaste, ghar). This distinction validates the linguistic features captured by our lexicon-based approach.

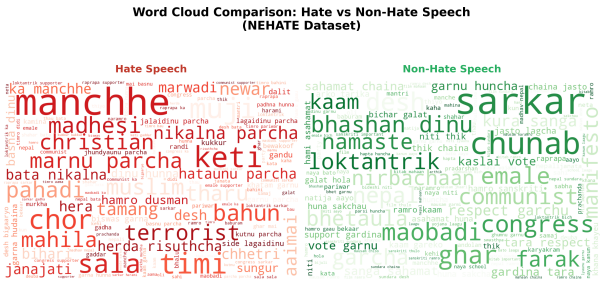

Fig. 3. Hate vs Non-Hate Word Comparison (NEHATE dataset)

C. Text Pre-processing

The preprocessing model handles both Devanagiri and Romanized Nepali text. The model first extracts raw data and removes URL, hashtags, symbols and normalize the characters for Devanagari...

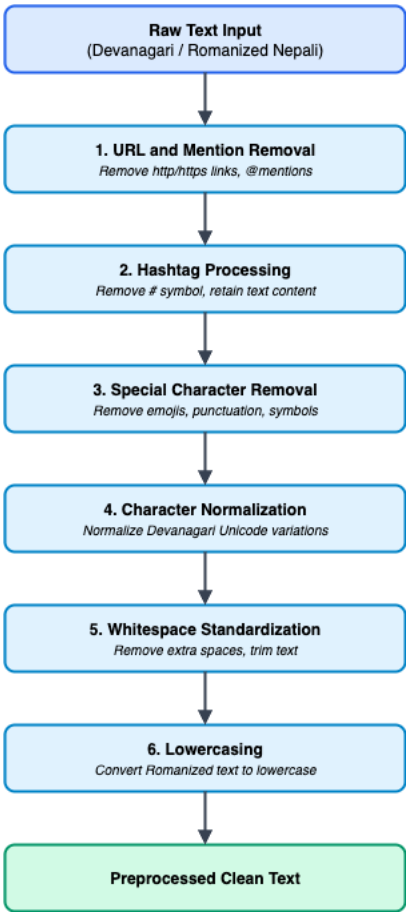

Fig. 4. Text Pre-processing

D. Feature Engineering

After pre-processing the text, two major types of features were extracted: character-level TF-IDF features and lexicon-based features.

1) Character-level TF-IDF: We utilized character-level n-grams(2-5) with TF-IDF weighting, which is a powerful and effective method for morphologically rich languages like Nepali. Character n-gram captures sub-word patterns, variations in code-mixed text common in social media.

2) Lexicon-based Features: In addition to statistical features, domain-specific linguistic features were derived using a curated Nepali offensive lexicon. Instead of using the lexicon as training data, we used it to construct feature signals that complement the TF-IDF representation. For each text sample, the following lexicon-based attributes were computed:

The offensive-terms-in-nepali dataset contained offensive terms, which we used for extracting features rather than training data (which would conflate term detection with hate speech detection).For each text sample, the following lexicon-based attributes were computed:

- Count of offensive terms present in the text.
- Maximum offensiveness score.
- Average offensiveness score across detected terms.
- Ratio of offensive terms to total words.
- Count of taboo terms.
- Category-specific term counts for semantic classes such as Politics, Race, Gender, Vulgarity, and Disability.

3) Feature Matrix Construction: The feature vectors generated from the character n-grams and the complete set of lexicon-based feature were stacked to form the final feature matrix. This strategic combination ensured the model received both general statistical linguistic information and domain-specific knowledge about offensive language.

E. Machine Learning Models

1) Random Forest Classifier: Random Forest Classifier is an ensemble learning method that constructs multiple decision trees during training and outputs the mode of classes for classification tasks. [9]

$$\hat{y} = \text{mode}\{h_t(x)\}_{t=1}^{T} \tag{1}$$

A total of 100 estimators trees provide a good balance combined with Gini impurity to compute for this binary class.

2) Naive Bayes: Naive Bayes method is a supervised learning algorithms based on Bayes' theorem, with the naive assumption that all features are conditionally independent given the class variable. The posterior probability of class C given a feature vector $\mathbf{X} = (x_1, x_2, \ldots, x_n)$ is:

$$P(C \mid \mathbf{X}) \propto P(C) \prod_{i=1}^{n} P(x_i \mid C), \tag{2}$$

where $P(C)$ is the prior probability of class C, and $P(x_i \mid C)$ is the conditional probability of feature x_i given class C [10].

The alpha value 0.1 is used to provide mild smoothing to handle zero frequency terms. Lower alpha values work better for text classification with sparse TF-IDF features.

3) Gradient Boosting: Gradient Boosting is an ensemble learning algorithm where each new model is trained to minimize the loss function (e.g., mean squared error or cross-entropy) of the previous ensemble using gradient descent. For N iterations, the final prediction $\hat{y}$ is:

$$\hat{y} = y_1 + \eta \sum_{i=1}^{N} r_i, \tag{3}$$

where y_1 is the initial prediction, r_i is the residual (error) predicted by the i-th weak learner, and η is the learning rate [11].
The no. of boosting stages is set at 100 with learning rate 0.1 upto max-depth 5 which prevents individual trees from being too complex.

F. Transformer Models

1) MuRIL: MuRIL (Multilingual Representations for Indian Languages) is a BERT-based model pre-trained on 17 Indian languages and their transliterated counterparts. The pre-trained model includes the masked language modeling (MLM) layer, allowing masked word prediction. An additional pre-processing module is used to convert raw text into the input format expected by the encoder [12].
For fine tuning, the learning rate was set to $2e^{-}5$ with a batch size of 16 for training to a max epoch of 5 with maximum sequence length 128 with adam optimizer. The binary cross entropy is used for classification with a dropout value of 0.1. Also, the L2 regularization coefficient is set at 0.01.

2) XLM-RoBERTa: XLM-RoBERTa is a large multilingual masked language model trained on 2.5TB of filtered CommonCrawl data across 100 languages. It shows that scaling the model provides strong performance gains on high-resource and low-resource languages. The model uses the RoBERTa pretraining objectives on the XLM model [13].
For fine tuning, the learning rate was set to $2e^{-}5$ with a batch size of 16 for training to a max epoch of 5 with maximum sequence length 128 with adam optimizer.

3) mBERT: The multilingual BERT (mBERT) model is pretrained on the top 104 languages using Wikipedia data with a masked language modeling (MLM) objective. BERT is a transformer model pretrained in a self-supervised manner, using only raw text without human labels. Automatic input-label generation allows it to leverage large-scale multilingual data efficiently.

V. Results and Discussion

All the experiments were conducted using an 80-20 stratified train-test split (10,804 training and 2701 test samples) with random_state 42 for reproducibility. The comprehensive performance evaluation is done below:

A. Traditional Machine Learning Results

Table I represents the performance of traditional ML models using combined TF-IDF and lexicon features.

TABLE I
TRADITIONAL ML MODEL EVALUATION

Model	***Accuracy***	***F1-Score***	*AUC-ROC*
Naive Bayes	0.824	0.821	0.889
Gradient Boosting	0.839	0.835	0.901
Random Forest	0.852	0.847	0.912

Among Traditional ML models, Random Forest achieves the highest performance with an F1-score of 0.847 and AUC-ROC of 0.912. The ensemble nature of Random Forest proves effective for the high dimensional feature space combining TF-IDF and lexicon features. Naive Bayes despite its independence assumption, achieves good result. All of the ML models outperforms random baseline ($AUC \geq 0.88$), indicating that the combined feature approach captures meaningful patterns for hate speech detection.

B. Transformer Model Results

Table II presents the performance of fine-tuned transformer models.

TABLE II
TRANSFORMER BASED MODEL EVALUATION

Model	***Accuracy***	***F1-Score***	AUC-ROC
m-BERT	0.851	0.852	0.914
XLM-RoBERTa	0.843	0.848	0.908
MuRIL	0.867	0.863	0.908

MuRIL achieves the best overall performance with F1-score of 0.863 and AUC-ROC of 0.908, outperforming both mBERT and XLM-RoBERTa. This superior performance is attributed to MuRIL's pretraining on 17 Indian Languages including transliterated text, which better captures the linguistic patterns of Nepali and its Romanized variants common on social media. XLM-RoBERTa despite its larger training corpus, achieves slightly lower scores possibly due to its broader multilingual scope diluting language specific patterns.

C. Ablation Study

Table III presents the ablation study results using Random Forest to isolate feature contributions.

TABLE III
ABLATION STUDY(RANDOM FOREST)

Feature Set	***Accuracy***	***F1-Score***	*AUC-ROC*
TF-IDF only	0.812	0.827	0.876
Lexicon Only	0.724	0.689	0.781
TF-IDF + Lexicon	0.852	0.847	0.912

The ablation study quantifies the contribution of each feature type. TF-IDF alone achieve a F1-score of 0.827 that shows the effectiveness of character n-grams for Nepali text. Lexicon features yield lower performance cause the text may not contain the lexicon terms. However, combining both TF-IDF with lexicon feature improves F1-score to 0.847. This

validates that domain specific linguistic knowledge enhances statistical text representation for hate speech detection.

D. ROC Curve Analysis

Figure 5 shows the ROC curves for traditional ML methods comparing lexicon-based features only versus vs combined TF-IDF and lexicon features. The figure demonstrates that using lexicon features alone results in limited detection accuracy as many hate speech texts may not contain terms present in the lexicon. However, combining TF-IDF character n-grams with lexicon features significantly improves performance over lexicon-only features. This combined approach captures both explicit offensive terms through the lexicon and implicit linguistic patterns through TF-IDF.

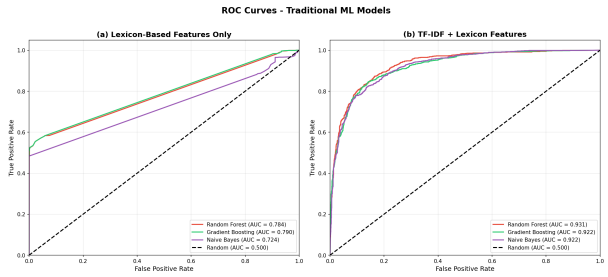

Fig. 5. ROC curve for ML method using lexicon only vs tf-idf with lexicon

VI. Conclusion and Future Works

This paper presented a comprehensive study on hate speech detection in Nepali social media. Traditional machine learning models using character-level TF-IDF performed strongly, with Random Forest Classifier giving the best results. Transformer based models also performed well, and MuRIL offered best performance for this task. The Nepali language, despite being a low-resource language, achieves a strong performance by combining lexicon-based feature engineering with modern classification approaches.

However, the study is limited by the size of the dataset and the scarcity of Nepali-specific pre-trained language models. Therefore, future work will focus on developing Nepali-specific pre-trained language models and expanding the dataset with more hate speech examples. Additionally, we aim to explore cross-lingual transmission from Hindi. The study also intends to investigate explainability techniques to better understand how models classify hate speech.

Acknowledgment

The authors thank the creators of the NEHATE dataset and the Nepali offensive terms lexicon for making their resources publicly available. We also acknowledge the HuggingFace team for providing accessible transformer model implementations.

References

[1] A. Sellars, “Defining hate speech,” Berkman Klein Center Research Publication.

[2] N. B. Niraula, S. Dulal, and D. Koirala, ”Linguistic Taboos and Euphemisms in Nepali,” ACM Transactions on Asian and Low-Resource Language Information Processing, vol. 21, no. 6, 2022. DOI: 10.1145/3524111

[3] D. Sharma, A. Singh, V. Singh, “Thar—targeted hate speech against religion: A high-quality Hindi-English code-mixed dataset with application of deep learning models for automatic detection,” ACM Transactions on Asian and Low Resource Language Information Processing, 2024.

[4] A. Guragain, N. Poudel, R. Piryani, B. Khannal, “NLPineers@ NLU of Devanagari Script Languages 2025: Hate Speech Detection using Ensembling of BERT-based Models,” in *Proc. of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025)*, 2025, pp. 320–326.

[5] D. Chakraborty, J. Hossain, M. Hoque, “NLU of Devanagari Script Languages 2025: Target Identification for Hate Speech Leveraging Transformer-based Approach,” in *Proc. of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025)*, 2025, pp. 327–333.

[6] J. Devlin, M. Chang, K. Lee, and K. Toutanova, ”BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” in Proc. NAACL-HLT, 2019, pp. 4171-4186.

[7] F. Pedregosa et al., ”Scikit-learn: Machine Learning in Python,” Journal of Machine Learning Research, vol. 12, pp. 2825-2830, 2011.

[8] A. Conneau et al., ”Unsupervised Cross-lingual Representation Learning at Scale,” in Proc. ACL, 2020, pp. 8440-8451.

[9] V.Kulkarni, ”Random Forest Classifiers: A Survey and Future Research Directions.” *International Journal of Advanced Computing*, vol.36, no.1, pp. 2051-0845, Apr.2013, Available: https://adiwijaya.staff.telkomuniversity.ac.id/files/2014/02/Random-Forest-Classifiers_A-Survey-and-Future.pdf

[10] Scikit-learn, “Naive Bayes,” https://scikit-learn.org/stable/modules/naive_bayes.html, Accessed: 2025.

[11] GeeksforGeeks, “Gradient Boosting — Machine Learning,” https://www.geeksforgeeks.org/machine-learning/ml-gradient-boosting/, Accessed: 2025.

[12] S. Khanuja et al., “MuRIL: Multilingual Representations for Indian Languages,” 2021. Available: https://arxiv.org/abs/2103.10730.

[13] Hugging Face Docs, “XLM-RoBERTa model documentation,” https://huggingface.co/docs/transformers/en/model_doc/xlm-roberta, Accessed: 2025.