DLRG-DravidianLangTech@EACL2024 : Combating Hate Speech in Telugu Code-mixed Text on Social Media

Ratnavel Rajalakshmi, Saptharishee M, Hareesh Teja S,
Gabriel Joshua R, and **Varsini SR**

School of Computer Science and Engineering
Vellore Institute of Technology, Chennai
Tamil Nadu, India
rajalakshmi.r@vit.ac.in

Abstract

Detecting hate speech in code-mixed language is vital for a secure online space, curbing harmful content, promoting inclusive communication, and safeguarding users from discrimination. Despite the linguistic complexities of code-mixed languages, this study explores diverse pre-processing methods. It finds that the Transliteration method excels in handling linguistic variations. The research comprehensively investigates machine learning and deep learning approaches, namely Logistic Regression and Bi-directional Gated Recurrent Unit (Bi-GRU) models. These models achieved F1 scores of 0.68 and 0.70, respectively, contributing to ongoing efforts to combat hate speech in code-mixed languages and offering valuable insights for future research in this critical domain.

1 Introduction

The surge in hateful speech online challenges maintaining respectful discourse. Hate speech, involving hostility or discrimination, has profound implications for social harmony. Digital platforms invest heavily in hate detection models to automate content flagging and removal, aiming to curb its spread. Addressing hate speech in Telugu code-mixed language is a growing concern due to the rapid adoption of digital platforms by the Indian population.

Automating hate speech detection is feasible for widely adopted languages like English, with ample models and labeled data. However, applying the same processes to niche languages like Telugu, Tamil, Malayalam, etc., remains unexplored due to complexities and nuances, making it a more expensive endeavor. The demand for automated hate speech detection in code-mix languages is underscored by the infeasibility of the conventional manual review approach for low-resourced languages in handling the vast amount of digital data.

In this study, two distinct models, the Bi-GRU and Logistic Regression, were carefully chosen to address the complexities of hate speech detection in code-mixed Telugu language. The Bi-GRU, a deep learning model, excels in capturing intricate contextual relationships, leveraging its ability to analyze sequences of data bidirectionally. This is particularly advantageous for understanding the nuanced linguistic structures present in code-mixed languages. On the other hand, Logistic Regression, a machine learning model, proves efficient in utilizing linguistic features, word embeddings, and statistical patterns. These models aims to harness the strengths of both paradigms, allowing for a comprehensive and nuanced approach to hate speech classification. These techniques reflects a thoughtful strategy to effectively tackle the multifaceted nature of hate speech detection in Telugu code-mixed languages in social media.

2 Related Works

Dealing with challenges in low-resource languages such as Dravidian languages involves addressing class imbalances as a major concern. These challenges were addressed by generating synthetic data through paraphrasing, utilizing the PEGASUS fine-tuned model, and employing backtranslation with the M2M100 neural machine translation model (Ganganwar and Rajalakshmi, 2023). A study on part-of-speech (POS) tagging for code-mixed English-Telugu social media text tackled challenges in combining elements from different languages. Classifiers like Linear SVMs, CRFs, and Multinomial Bayes, with varied feature combinations, were evaluated. CRF outperformed SVMs and Bayes classifiers in this context (Nelakuditi et al., 2018).

Hate speech and offensive content detection in Malayalam and Tamil code-mixed text used the

Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages, pages 140–145
March 22, 2024 ©2024 Association for Computational Linguistics

HASOC-FIRE 2021 dataset. The MuRIL model achieved the best performance with a weighted F1-score of 0.636 for Tamil and 0.734 for Malayalam (Bhawal et al., 2022). Advanced multilingual Transformer models, adopting a unique fine-tuning approach with learning rate scheduling based on macro F1-scores (Ghosh Roy et al., 2021), have shown success in identifying hate speech. The mBERT-GRU framework for hate speech detection in multilingual societies outperforms monolingual and state-of-the-art methods (Singh et al., 2023).

The rise of hate speech on social media calls for automated detection using NLP models. Integrating convolutional and recurrent layers yields 77.16% accuracy in identifying hate speech (Shubhang et al., 2023). Multinomial Logistic Regression for hate speech on Twitter achieves an average precision of 80.02%, recall of 82%, and accuracy of 87.68% (Br Ginting et al., 2019). Hate speech detection in Bengali comments, with a dataset of 7,425 comments, successfully addresses challenges. The attention mechanism surpasses other algorithms with 77% accuracy (Das et al., 2021). The exploration of abusive comment detection within the Tamil+English dataset involved the utilization of Random Forest, resulting in a weighted average F1-score of 0.78 (Rajalakshmi et al., 2022).

The Random Forest Classifier exhibited a notable performance in the Hate Speech and Offensive Content Identification in Marathi and Hindi tweet datasets by achieving a macro F1 score of 75.19% and 73.12% (Rajalakshmi et al., 2021). Earlier study (Rajalakshmi, 2014) explored term weighting methods aimed at selecting pertinent URL features and assessing their influence on the effectiveness of URL classification, extending beyond the realm of text classification . In the domain of multilingual social media content, a novel relevance-based metric was introduced through the application of a statistics-based approach, facilitating the swift processing of multilingual queries (Rajalakshmi and Agrawal, 2017). As a progressive phase in social media data analysis, multimodal face emotion recognition on code-mixed Tamil memes was conducted by applying Convolutional Neural Network (CNN) with an efficiency of 0.3028 (Kannan et al., 2023). For sentiment analysis, various deep learning methods were applied (Sivakumar and Rajalakshmi, 2021, 2022). In Tamil hate and offensive content identification, the role of stemming and stop words were analysed in (Rajalakshmi et al., 2023)

3 Methodology

3.1 Data Overview

The dataset used in this study is a part of the shared task (B et al., 2024) in Codalab (`https://codalab.lisn.upsaclay.fr/competitions/16095`). The task given was to identify hate content in Telugu code-mixed text (Priyadharshini et al., 2023). The dataset has training and testing sets, comprising of 4000 and 500 entries respectively. The composition of data is shown in Table 1. The near-equal distribution of labels minimises sampling biases, enhancing the reliability of the subsequent analysis.

Data	Hate	Non-Hate
Training data	2,061	1,939
Testing Data	250	250

Table 1: Dataset Statistics

3.2 Data Pre-processing

Text pre-processing in Telugu code-mixed comments is crucial for enhancing model performance, addressing language variations, removing noise, and ensuring consistency for accurate analysis. The process begins with comment cleaning by removing unnecessary white spaces and lines. AI4Bharat Indic-Transliteration (Madhani, 2022) was employed for transliteration, converting text from one script to another without focusing on meaning of the translation. Transliteration aids hate speech identification in code-mixed Telugu on social media by converting mixed-script content to a uniform script. This process ensures consistent language representation, facilitating more effective and accurate detection of offensive language patterns. Post-transliteration, additional cleaning removes non-alphanumeric characters, and the text is converted to lowercase for uniformity. Tokenization using the Natural Language Toolkit (NLTK) follows standardized text processing. For machine learning models, Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer extracts features, while deep learning models utilize tokenization and sequence padding to ensure consistent input sequence lengths.

3.3 Model Building

In this study, two different classification algorithms have been applied and the details are presented below.

3.3.1 Logistic Regression

Logistic regression is a common statistical classifier for binary classification tasks, utilizing a sigmoid function to transform the linear combination of input features. This mapping, ranging between 0 and 1, represents the probability of an instance belonging to the positive class. Default parameters, including L2 regularization (C = 1.0) to prevent over-fitting and the ’lbfgs’ solver, were employed for hate speech classification. These defaults strike a balanced trade-off between model complexity and generalization, suitable for small to medium-sized datasets in logistic regression tasks.

The logistic regressor is represented as –

$$P(y = 1) = \frac{1}{1 + e^{-(w_1x_1+w_2x_2+\ldots+w_nx_n+c)}} \quad (1)$$

where $P(y = 1)$ is the probability of the instance belonging to the positive class, $x_1, \ldots, x_n$ are the input features, $w_1, \ldots, w_n$ are the weights assigned to each feature, and c is the bias.

3.3.2 Bidirectional Gated Recurrent Unit (Bi-GRU)

Bi-GRU, a variant of RNNs, uses gating mechanisms to control information flow. With two gates (update and reset) capturing contextual information bidirectionally, it enhances understanding of sequential dependencies. This bidirectional nature improves classification accuracy and overall performance by enabling the model to grasp nuanced relationships within the text.

The architecture of Double cell Bi-GRU model has an input layer which processes a 71-feature sequence vector, followed by an embedding layer. Two Bi-GRU layers with 128 and 64 neurons capture intricate patterns. Three dense layers use ReLU activation (64 and 32 neurons), and the final output layer employs sigmoid activation. The model uses the Adam optimizer with default settings, binary cross-entropy loss function, and trains for 5 epochs with a batch size of 32. This design ensures effective learning while maintaining computational efficiency in the Bi-GRU model.

4 Results and Discussion

Both proposed models for the classification task were studied and the results are discussed below. From Table 2, we can observe that Bi-GRU outperforms Logistic Regression in training accuracy (99.6% vs. 92.9%), indicating superior fitting to the training data. However, during testing, Bi-GRU’s accuracy (69.4%) only slightly surpasses Logistic Regression (68.2%). Despite significantly lower training loss for Bi-GRU (0.014) compared to Logistic Regression (0.418), its testing loss (1.143) is higher than Logistic Regression (0.612), suggesting potential over-fitting. In summary, while Bi-GRU excels in training accuracy and loss, both models exhibit similar testing accuracy, with Logistic Regression demonstrating slightly better generalization performance.

Model	Bi-GRU	Logistic Regression
Training Accuracy	0.996	0.929
Testing Accuracy	0.694	0.682
Training Loss	0.014	0.418
Testing Loss	1.143	0.612

Table 2: Comparison of Performance

Class	Precision	Recall	F1-Score
Hate	0.68	0.70	0.69
Non-hate	0.69	0.67	0.68
Accuracy			0.68
Macro Avg	0.68	0.68	0.68
Weighted Avg	0.68	0.68	0.68

Table 3: Logistic Regression Classification Report

The classification report of models are shown in Table 3 and Table 4. The Bi-GRU model outperforms logistic regression. Logistic regression achieves balanced precision (0.68) and recall (0.70 for hate, 0.67 for non-hate) with F1-scores of 0.69 and 0.68, and contributing to an overall accuracy of 0.68. In comparison, Bi-GRU excels with precision (0.68 for hate, 0.71 for non-hate) and a high recall of 0.74 for hate. F1-scores for "Hate" and "Non-hate" are 0.71 and 0.68, respectively, culminating in an overall accuracy of 0.70, highlighting the model’s performance across both classes.

Class	Precision	Recall	F1-Score
Hate	0.68	0.74	0.71
Non-hate	0.71	0.65	0.68
Accuracy			0.70
Macro Avg	0.70	0.70	0.70
Weighted Avg	0.70	0.70	0.70

Table 4: Bi-GRU Classification Performance

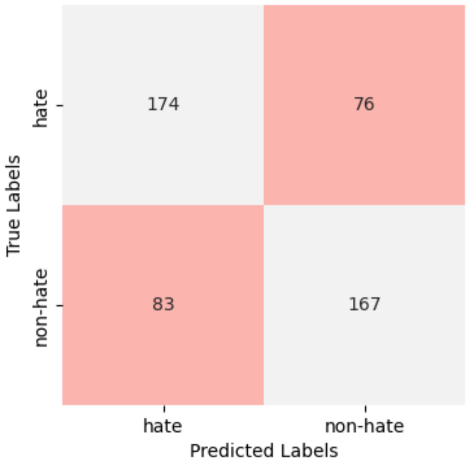

Figure 1: Logistic Regression Model - Confusion Matrix

The confusion matrix of Logistic regression and Bi-GRU are illustrated in Fig. 1 and Fig. 2. In Bi-GRU, 185 out of 250 hate comments were correctly classified, compared to 174 by Logistic Regression. For non-hate comments, Logistic Regression accurately classified 167, while bi-GRU correctly classified 163 out of 250. Although misclassifications are limited in both models, fine-tuning could further enhance performance.

A Receiver Operating Characteristic (ROC) plots visually showcase a binary classification model's ability to distinguish between classes across various threshold values. Fig. 3 and Fig. 4 depict the trade-off between sensitivity and specificity for logistic and Bi-GRU models. A curve closer to the top-left corner indicates superior discrimination compared to random chance (diagonal line). The Area Under the Curve (AUC) summarizes overall performance, and a higher AUC reflects better discrimination for both models in the hate speech detection task.

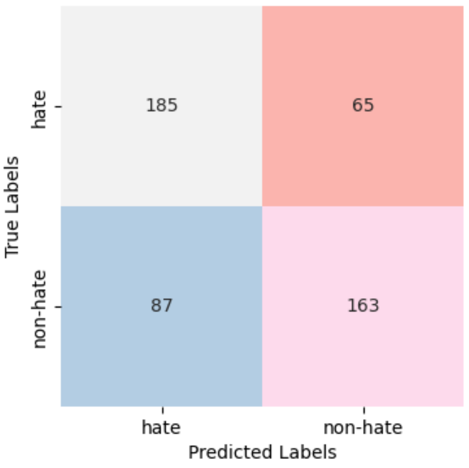

Figure 2: Bi-GRU Model - Confusion Matrix

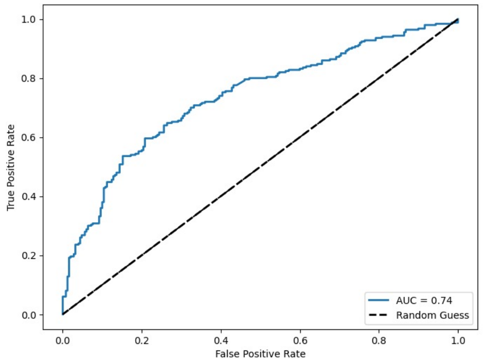

Figure 3: Logistic Regression - ROC Curve

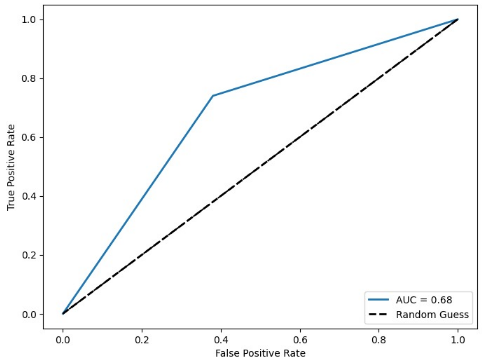

Figure 4: Bi-GRU - ROC Curve

Reduction in accuracy observed with incorporating regularization and dropout methods can be attributed to the dataset's limited size, leading to under-fitting. With a small dataset, regularization may hinder model complexity, exacerbating under-fitting issues. To enhance accuracy, the pragmatic approach involves adding more layers, compromising on simplicity but addressing the under-fitting challenge. The paper's suggestion of regularization methods aligns with the need for improved model generalization, yet the dataset's size necessitates a nuanced trade-off, favoring increased model complexity for enhanced performance on the limited training data.

Name	Score	Rank
Sandalphon	0.7711	1
Selam	0.7711	1
Kubapok	0.7431	3
DLRG1	0.7101	4
DLRG	0.7041	5
CUET_Binary_Hackers	0.7013	6
CUET_OpenNLP_HOLD	0.6878	7
Zavira	0.6819	8
IIITDWD-zk_lstm	0.6739	9
lemlem - Moein Tash	0.6708	10

Table 5: Ranklist of HOLD-Telugu

The outcomes of the Hate and Offensive Language Detection in Telugu code-mixed Text (HOLD-Telugu) Shared task of Codalab competition are presented in Table 5. Our proposed model achieved the 5th position, demonstrating exceptional performance attributed to its effective handling of code-mixed Telugu through transliteration. This critical step involved in mitigating variation in code-mixed text significantly contributed to the model's success. Furthermore, the employed methods, logistic regression along with word embedding, and Bi-GRU bidirectional sequence analysis, have proven to be effective in handling code-mixed Telugu language and accurately classifying them. Therefore, the ultimate goal of detecting and eliminating hate speech from social media, contributing to building a safe and inclusive digital society, has been achieved.

Future work involves expanding data collection across diverse platforms and regions to enhance dataset representativeness. Employing data augmentation techniques, such as oversampling and synthetic data generation, will address class imbalances. Implementing a data curation strategy is crucial to mitigate biases and ensure ethically sound models. Exploring alternative deep learning architectures aims to enhance overall model performance. Additionally, integration of the model into real-time systems on social media platforms will enable swift intervention against hate speech, contributing to a safer online environment.

5 Conclusion

Classifying code-mixed, low-resource Dravidian languages like Telugu in social media is challenging due to the availability of limited labeled data, diverse language variations, and informal expressions. Ambiguous language use and the absence of standardized resources make building effective models difficult, requiring tailored approaches for accurate sentiment and content analysis. Logistic regression and Bi-GRU for Telugu code-mix hate classification effectively capture complex patterns, enhancing contextual understanding for nuanced hate speech detection in Telugu code-mix. Refining fine-tuning and pre-processing techniques can further improve model efficacy.

Limitations

Despite the valuable insights provided by the dataset, its small size may limit the model's representation of online discourse, potentially impacting overall robustness. The presence of potential class imbalances within specific hate speech types could hinder accuracy, and inherent biases in the data based on social and cultural perspectives might result in unfair detection. The constrained model architecture may benefit from exploration of advanced approaches tailored for code-mixed languages. Transliteration errors introduced by IndicXlit-AI4Bharath further challenge the model's understanding of Telugu nuances. Additionally, relying solely on individual comments disregards surrounding context, affecting sarcasm and irony detection. This section underscores the need for continued research to address these limitations and advance the model's effectiveness in diverse linguistic and contextual scenarios.

Ethics Statement

This study on hate speech detection in code-mixed languages aligns with ACL's Ethics Policy, upholding principles of integrity and responsibility. We emphasize the significance of fostering a secure online environment and mitigating harmful content. Adhering to ethical considerations, we explore diverse pre-processing methods, identifying the Transliteration approach as effective in handling linguistic complexities. Our research delves into machine learning methods, presenting Logistic Regression and Bi-GRU models with F1 scores of 0.68 and 0.70. The ethical impact of our work is acknowledged, and we encourage further discourse on its societal implications. This statement, post-conclusion, reflects our commitment to transparency and responsible research, contributing to ethical standards in scientific inquiry.

References

Premjth B, Bharathi Raja, Prasanna Kumar Kumaresan, Saranya Rajiakodi, Sai Prashanth Karnati, Sai Rishith Reddy Mangamuru, and Janakiram Chandu. 2024. Findings of the shared task on Hate and Offensive Language Detection in Telugu code-mixed text (HOLD-Telugu).

Snehaan Bhawal, Pradeep Roy, and Abhinav Kumar. 2022. Hate speech and offensive language identification on multilingual code-mixed text using BERT.

P. S. Br Ginting, B. Irawan, and C. Setianingsih. 2019. Hate speech detection on twitter using multinomial logistic regression classification method. pages 105–111.

A. Das, A. Al Asif, A. Paul, and M. Hossain. 2021. Bangla hate speech detection on social media using attention-based recurrent neural network. *Journal of Intelligent Systems*, 30(1):578–591.

V. Ganganwar and R. Rajalakshmi. 2023. Employing synthetic data for addressing the class imbalance in aspect-based sentiment classification. *Journal of Information and Telecommunication*, pages 1–22.

S. Ghosh Roy, U. Narayan, T. Raha, Z. Abid, and V. Varma. 2021. Leveraging multilingual transformers for hate speech detection. *ArXiv*.

R. R. Kannan, M. Ravikiran, and R. Rajalakshmi. 2023. MMOD-Meme: A dataset for multimodal face emotion recognition on code-mixed Tamil memes. *Communications in Computer and Information Science*, pages 335–345.

Y. Madhani. 2022. Aksharantar: Open Indic-language transliteration datasets and models for the next billion users. *arXiv.org*.

K. Nelakuditi, D. S. Jitta, and R. Mamidi. 2018. Part-of-speech tagging for code-mixed english-telugu social media data. 9623:578–591.

Ruba Priyadharshini, Bharathi Raja Chakravarthi, Malliga S, SUBALALITHA CN, Kogilavani SV, Premjith B, Abirami Murugappan, and Prasanna Kumar Kumaresan. 2023. Overview of Shared-Task on Abusive Comment Detection in Tamil and Telugu.

R. Rajalakshmi. 2014. Supervised term weighting methods for URL classification. *Journal of Computer Science*, 10(10):1969–1976.

R. Rajalakshmi and R. Agrawal. 2017. Borrowing likeliness ranking based on relevance factor.

R. Rajalakshmi, A. Duraphe, and A. Shibani. 2022. DLRG@DravidianLangTech@2022: Abusive comment detection in Tamil using multilingual transformer models. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*.

R. Rajalakshmi, F. Mattins, S. Srivarshan, P. Reddy, and M. A. Kumar. 2021. Hate speech and offensive content identification in Hindi and Marathi language tweets using ensemble techniques. *Fire*.

R. Rajalakshmi, S. Selvaraj, F. M. R., P. Vasudevan, and A. K. M. 2023. Hottest: Hate and offensive content identification in tamil using transformers and enhanced stemming. *Computer Speech & Language*, 78:101464.

S. Shubhang, S. Kumar, U. Jindal, A. Kumar, and N. R. Roy. 2023. Identification of hate speech and offensive content using BI-GRU-LSTM-CNN Model. pages 536–541.

P. Singh, N. Singh, and S. Chand. 2023. mbert-gru multilingual deep learning framework for hate speech detection in social media. *Journal of Intelligent and Fuzzy Systems*, 44(5):8177–8192.

S. Sivakumar and R. Rajalakshmi. 2021. Self-attention based sentiment analysis with effective embedding techniques. *International Journal of Computer Applications in Technology*, 65(1):65.

S. Sivakumar and R. Rajalakshmi. 2022. Context-aware sentiment analysis with attention-enhanced features from bidirectional transformers. *Social Network Analysis and Mining*, 12(1).