A Transformer-based model for hate speech detection in low resource Indian languages

Jaimin Damor[1], Siba Sankar Sahu[1]

[1]*Department of Computer Science and Engineering, Sardar Vallabhbhai National Institute of Technology, Surat, India*

Abstract

Hate speech detection is a critical challenge in low-resource Indian languages, where people express their opinions in several languages on social networks. As part of the FIRE 2025 HASOC shared task, our goal is to detect hate speech in different Indian languages including Bangla, Hindi, Gujarati, and Bodo by focusing on four subtasks: Sentiment, Sarcasm, Vulgarity, and Abuse detection. We leverage pre-trained transformer models including XLM-ROBERTa, IndicBERT, MuRIL, and mBERT enhanced with convolutional neural network (CNN) layers in different Indian languages. Experimental results show that MuRIL provides optimal performance and outperforms other transformer models in low-resource Indian languages. Our team CSE_SVNIT achieving a competitive macro F_1-scores of 0.5547 (Bangla, Rank 10/17), 0.5719 (Hindi, Rank 7/18), 0.5822 (Gujarati, Rank 9/15) and 0.5873 (Bodo, Rank 6/15). The explored model offers competitive performance to other submitted models in the HASOC shared task. The efficiency of MuRIL architecture for low-resource Indian languages suggests the integration of the model into real-life scenarios to provide a safer online environment.

Keywords

Hate speech detection, Indian languages, Multilingual NLP, Text analysis, Transformer models

1. Introduction

The proliferation of user-generated content on social media platforms like Twitter, Facebook, Instagram, and YouTube revolutionized communication, allowing more than 4.7 billion users to share thoughts, expressions, and engage in real-time interactions [1]. Digital transformation has allowed individuals to express diverse perspectives; however, few people spread hate speech, offensive remarks, and toxic content, posing significant threats to social harmony and democratic discourse [2]. In India, with a population of 1.4 billion, people speak various languages such as Hindi, Gujarati, Marathi, Bangla, and Bodo during communication on social networks. The challenge of detecting hate speech in low-resource Indian languages was due to the less availability of linguistic resources on the Web.

The HASOC[2](Hate Speech and Offensive Content Identification) [32], [33] shared task is a part of the FIRE [3] (Forum for Information Retrieval Evaluation) evaluation campaign introducing a hate speech detection in various low-resource Indian languages. They conducted several downstream tasks such as hate speech detection [14], [24], hate speech multiclass classification [12], [25], and categorization of offensive content [16], [31]. In the HASOC 2025 shared task, they performed a multimodal hate speech detection in low-resource Indian languages. They primarily considered four low-resource Indian languages, like Bangla, Hindi, Gujarati, and Bodo. The task focuses on identifying nuanced harmful content through four subtasks: Sentiment detection, Sarcasm detection, Vulgarity detection, and Abuse detection [34]. Focusing on the four subtasks, we developed a robust framework that improves hate speech detection accuracy and establishes the foundation model for underrepresented languages. Although the task is multimodal, we adopt a purely text-based approach, focusing solely on the textual content of the memes.

Forum for Information Retrieval Evaluation, December 17-20, 2025, India

p24ds020@coed.svnit.ac.in (J. Damor); sibasankar@coed.svnit.ac.in (S. S. Sahu)
https (J. Damor); https (S. S. Sahu)
0009-0006-6531-2234 (J. Damor); 0000-0001-9769-9206 (S. S. Sahu)

[1] https://datareportal.com/reports/digital-2025-global-overview-report
[2] https://hasocfire.github.io/hasoc/2025/
[3] https://fire.irsi.org.in/fire/2025/home

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-02-24

In recent years, transformer models have been used in the detection of hate speech content in low resource languages [11], [26]. These models offer optimal performance and improved effectiveness in different downstream tasks, such as text classification [13], [20], hate speech detection [12], [24], sentiment analysis [8], [17]. In this study, we explore the effectiveness of transformer models for the detection of hate speech in multilingual, resource-constrained settings. Our methodology includes preprocessing, oversampling to address class imbalance, and fine-tuning to improve classification performance. The proposed model efficiently detects hate speech content in different low-resource languages and provides a safer digital environment.

2. Related Work

Hate speech detection is a crucial downstream task in natural language processing due to the widespread dissemination of offensive and harmful content across online platforms. Although significant research has focused on the detection of hate speech in European languages, particularly English [24], exploration in low-resource Indian languages remains limited. Researchers have explored a variety of models, from traditional machine learning to currently transformer-based models to identify hate and offensive content on the Web. We outline key techniques and their applications below.

2.1. HASOC Shared Task

The design, dataset construction, and evaluation protocol of the HASOC 2025 shared task are comprehensively described in the official overview papers [32], [33]. The HASOC (Hate Speech and Offensive Content Identification) track on FIRE has been a cornerstone for advancing hate speech detection in low-resource languages. Initiated in 2019 [4], it was initially aimed at Hindi, German, and English. The 2023 edition expanded to include Assamese, Bengali, Bodo, Gujarati, and Sinhala [21], [30]. In 2025, HASOC introduced abusive meme identification in Bangla, Hindi, Gujarati, and Bodo. These shared tasks have provided standardized datasets and evaluation frameworks, enabling fair comparison between methods.

2.2. Machine Learning Methods

Traditional machine learning techniques, including Support Vector Machines (SVM), Naive Bayes, Random Forest, Logistic Regression, and Decision Trees, remain widely used due to their interpretability and effectiveness with lexical features like TF-IDF and n-grams. Kumar et al. [19] leveraged trigram features with Random Forest for Hindi hate speech detection, achieving a macro F_1-score of 0.82. Patel et al. [29] applied logistic regression and Naive Bayes on code-mixed Gujarati text. They found that Logistic Regression provides optimal performance and achieved a precision of 0.79. Gupta et al. [10] used Random Forest with TF-IDF on the HASOC 2022 Bangla dataset, achieving 83% accuracy. Singh et al. [23] evaluated SVM with n-gram features in Dravidian languages, obtaining macro F_1-scores of 0.75 (Tamil) and 0.68 (Malayalam).

2.3. Deep Learning Methods

Deep learning models address the limitations of some traditional approaches by capturing long-range dependencies and contextual patterns. Das et al. [7] proposed an encoder–decoder architecture with 1D convolutional layers and an attention-based LSTM decoder for Bengali hate speech classification. They achieve an accuracy of 77% in different categories. Gupta et al. [28] implemented CNN and LSTM with word embeddings for Hindi, achieving a macro F_1-score of 0.81 and outperform the n-gram-based approach. Rao et al. [22] developed a hybrid BiLSTM-CNN model. The model offers superior performance on HASOC 2023 Hindi datasets.

2.4. Transformer-based Methods

In recent years, transformer-based models have been frequently used for different NLP tasks. Conneau et al. [6] introduced cross-lingual pre-training models like XLM-R to learn shared representations in more than hundred languages, making them ideal for low-resource languages. Patel et al. [29] found RoBERTa achieved a macro F_1-score of 0.85, outperforming traditional models. Rao et al. [22] reported XLM-RoBERTa providing a macro F_1 score of 0.82 in HASOC 2023 Hindi. Singh et al. [31] used XLM-RoBERTa for code-mixed Punjabi-English. They achieved a macro F_1 score of 0.77. Gupta et al. [18] applied DistilBERT variants to code-mixed Gujarati-English. They found a macro F_1 and a precision of 0.76 and 0.80. Das and Mukherjee [15] introduced a dataset of Bangla abuse memes for multimodal hate speech detection. They used data bootstrapping techniques to improve abusive language detection in low-resource Indic languages [9]. Ghosh et al. [27] developed a SafeSpeech pipeline to mitigate hate speech content in Indic social media texts. Despite these advances, challenges persist in multimodal, code-mixed, and low-resource Indian languages. In this study, we explore different transformer-based models like IndicBERT, MuRIL, XLM-RoBERTa, and mBERT for hate speech detection in Bangla, Hindi, Gujarati, and Bodo under the HASOC 2025 shared task.

The remainder of the paper is organized as follows. Section 3 provides the statistics of the datasets used in the HASOC shared task. The model framework is presented in Section 4, followed by the experimental results in Section 5. Finally, we conclude with future work in Section 6.

3. Dataset

The HASOC [32], [33][4] (Hate Speech and Offensive Content Identification) shared task was part of the FIRE [5] (Forum for Information Retrieval Evaluation) evaluation campaign. In HASOC 2025, the organizers provided four datasets: Bangla, Hindi, Gujarati, and Bodo for the detection of hate speech. The dataset was divided into training, validation, and test subsets. The training and validation sets were used for model development, while the test set was reserved for final evaluation. The annotation scheme aligns with the multilingual hate speech detection frameworks [14]. The statistic of the data is shown in Table 1. Each task was divided into four subtasks: Sentiment detection (positive, neutral, negative), Sarcasm detection (sarcastic, non-sarcastic), Vulgarity detection (vulgar, not vulgar), and Abuse detection (abusive, non-abusive). The statistics of each subtask are presented in Table 2.

Language	Total Samples	Training Samples	Validation Samples	Test Samples
Bangla	2,693	2,154	539	1,821
Hindi	1,141	913	228	769
Gujarati	889	711	178	604
Bodo	378	302	76	254

Table 1
Statistics of training, validation, and test samples in different Indian languages

4. Methodology

The hate speech detection model was implemented in three steps: data pre-processing, model design, and evaluation. The preprocessing steps included handling missing data, data normalization, tokenization, and label encoding. The model framework comprised transformer models with convolutional neural

[4] https://hasocfire.github.io/hasoc/2025/
[5] https://fire.irsi.org.in/fire/2025/home

Language	Sentiment			Sarcasm		Vulgar		Abuse	
	Positive	Negative	Neutral	Sarcasm	Non Sarcasm	Vulgar	Non Vulgar	Abusive	Non-Abusive
Bangla	1476	906	311	2081	612	467	2226	739	1954
Hindi	340	525	276	770	371	377	764	307	834
Gujarati	404	291	194	670	219	297	592	168	721
Bodo	227	151	0	339	39	107	269	77	301

Table 2
Statistics of each task in different Indian languages

network (CNN) architecture. Finally, we evaluated the effectiveness of the model using the macro-F_1 score.

4.1. Preprocessing

To ensure data quality, consistency, and compatibility with transformer models, we implement the following preprocessing steps in different Indian languages.

- **Handling missing or Invalid data:** Rows with missing or invalid OCR text were removed to ensure data integrity. Entries with fewer than 3 characters were filtered out and invalid placeholders such as #NAME? were excluded to focus on meaningful content, a practice supported by Brownlee's NLP data cleaning guidelines [3].
- **Normalization:** Text was processed to enhance uniformity through the following steps:
  - Convert to lowercase for consistency, though this is less critical for nonlatin scripts like Gujarati.
  - Special characters, numbers, and punctuation were removed, retaining language-specific unicode ranges (e.g., for Gujarati, adjusted for other languages) to preserve linguistic integrity.
  - Extra whitespace was consolidated to streamline the text.
  - Unicode normalization was applied using language-specific normalizers from the IndicNLP [6] library (e.g., for Gujarati, Hindi, etc.) to handle diacritics and script variations, aligned with multilingual preprocessing techniques by Conneau et al. (2020) [6].
- **Tokenization**: The IndicNLP library was used with language-specific settings (e.g., for Gujarati, Hindi) to split text into tokens, preserving morphological and syntactic features. The tokens were rejoined with spaces to maintain compatibility with transformer tokenizers, a method validated for Indian languages by Ghosh et al. (2023) [16].
- **Label encoding**: Categorical labels were converted to numerical values using predefined mappings:
  - Sentiment: `{'Positive':1, 'Negative':0, 'Neutral':2}`
  - Sarcasm: `{'Sarcastic':1, 'Non-sarcastic':0}`
  - Vulgar: `{'Vulgar':1, 'Non vulgar':0}`
  - Abuse: `{'Abusive':1, 'Non-abusive':0}`
  - Target: `{'Gender':0, 'Religion':1, 'Individual':2, 'Political':3, 'National origin':4, 'Social sub-groups':5, 'Others':6, 'None':7}`

  Missing `Target` values were filled with 7 ('None').

[6] https://github.com/anoopkunchukuttan/indic_nlp_library

4.2. Oversampling

Class imbalance is prevalent in hate speech datasets, with minority classes (e.g., abusive, vulgar) underrepresented. We address the issue using `resample` from scikit-learn to oversample minority classes to match the size of the majority class (e.g., nonabusive) within each preprocessed dataset. The approach involves random sampling with replacement, duplicating minority class instances to balance the dataset. It ensures balance training data, enhances model performance on imbalanced subtasks such as abuse detection, a technique recommended by Brownlee [3] for handling skewed datasets, and validated in multilingual contexts by Rajalakshmi and Reddy [5].

4.3. Model framework

We used four pre-trained transformer models such as MuRIL, IndicBERT, mBERT, and XLM-ROBERTa. MuRIL and IndicBERT specifically trained in Indian language datasets, including code-mixed and low-resource languages like Bangla, Hindi, Gujarati, and Bodo, which align with our datasets. mBERT is trained on a vast multilingual corpus, and XLM-RoBERTa, optimized for cross-lingual understanding, and leverage their ability to handle linguistic variations. Moreover, transformer-based models have the capability to capture long-range dependencies and contextual nuances in a sentence effectively. Hence, we explore the following transformer-based models in different low-resource Indian languages. A brief overview of each model is presented below.

- **mBERT**: Multilingual BERT (mBERT) is a variant of the Bidirectional Encoder Representations from Transformers, featuring a 12-layer architecture with 768 hidden units per layer and a shared multilingual vocabulary trained on a vast corpus of 104 languages. Its bidirectional training approach allows it to capture contextual relationships from both directions, making it adaptable to diverse linguistic structures. The model employs a wordpiece tokenizer to handle subword units, ensuring flexibility across varied language morphologies, and includes 12 attention heads per layer to enhance contextual understanding.
- **MuRIL**: Multilingual representations for Indian languages (MuRIL) is a transformer model with a 12-layer structure, designed with a focus on Indic scripts and code-mixed text, pre-trained on a comprehensive corpus of 17 Indian languages. It incorporates a language-agnostic tokenization strategy and a custom embedding layer to handle the phonetic and syntactic diversity of languages including Hindi, Gujarati, and Bangla. The model uses a dual encoder setup to process both language-specific and shared representations, supported by a vocabulary of more than 2,00,000 tokens, tailored to capture the complexity of Indic writing systems.
- **IndicBERT**: IndicBERT is a specialized transformer model with a 12-layer architecture, fine-tuned on a dataset of 12 major Indian languages, featuring a vocabulary tailored to capture the morphological richness of Indic scripts. Its design includes additional pretraining steps to enhance its sensitivity to language-specific patterns, particularly in resource-constrained settings. The architecture uses a sentencepiece tokenizer for efficient subword segmentation. The model integrates 16 attention heads per layer to refine contextual focus, optimizing it for the nuances of languages like Bodo and Gujarati.
- **XLM-ROBERTa**: The cross-lingual language model (XLM-RoBERTa) is an advanced transformer with a 24-layer architecture and 1024 hidden units. The model is pre-trained on 100 languages using an optimized training objective that includes dynamic masking and larger batch sizes. It employs a robust tokenizer and enhanced contextual understanding, enabling it to process a wide range of linguistic variations effectively, with 16 attention heads per layer to deepen cross-lingual feature extraction. The model training uses a byte-pair encoding strategy, expanding its vocabulary to more than 2,50,000 tokens, and includes a sophisticated normalization process to improve text representation across diverse scripts.

These models are fine-tuned with CNN layers to capture contextual and semantic features from text embeddings.

4.3.1. Parameter

The model architecture integrates transformer and CNN components for enhanced classification. The transformer model uses the following parameter for the detection of hate speech.

- **Tokenizer**: The text was tokenized using a pretrained Hugging Face tokenizer (e.g. `google/muril-base-cased` for MuRIL) with a maximum length of 128 tokens, adding special tokens and applying padding or truncation as needed, a standard practice in transformer-based NLP as described by Goodfellow et al. (2016) [1].
- **Transformer Backbone**: A pretrained transformer model (e.g., for MuRIL) generated contextual embeddings with 768 dimensions, serving as the foundation for text classification.
- **CNN Enhancement**: A custom CNN architecture integrated with the transformer backbone, employing a 1D convolutional layer with 768 input channels, 128 output channels, and a kernel size of 3, followed by ReLU activation, max pooling with a kernel size of 2, and a fully connected layer. A dropout rate of 0.3 is applied to prevent overfitting. The forward pass adjusts the embedding dimensions based on the sequence length (e.g., reduced by convolution and pooling), tailored to 2 to 3 classes per subtask. The hybrid approach is implemented by combining convolutional feature extraction with transformer embeddings, inspired by Zhang et al. [25].
- **Optimization**: The AdamW optimization method with a learning rate of 2e-5 was utilized, by its adaptive moment estimation properties.
- **Loss function**: Cross-entropy loss served as the primary objective function, with experiments employing focal loss (with gamma = 1.5 and alpha = 0.5) to address class imbalance. The technique applied to handle imbalanced datasets.

4.4. Experimental Setup

The training data set for each language was divided into training and validation sets using a stratified approach. We divided the data set into 80 percent training and 20 percent validation aligns with best practices in natural language processing [1]. The split is applied to each dataset file, utilizing the OCR column as a text input and the labels as a output. The stratified split strategy ensures that the proportion of classes is maintained in both the training and the testing sets, addressing the possible class imbalance for effective machine learning evaluation [3]. We divided the official training set into 80% training and 20% validation and applied the trained models to the official masked test dataset for final evaluation.

- **Environment**: Models were trained in Google Colab with GPU support to handle computational demands, with fallback to the CPU if runtime errors occur (e.g. memory issues), a practical approach for resource-constrained deep learning task [3].
- **Hyperparameters**: We used a batch size of 16, a maximum sequence length of 128, and the model was trained for 10 epochs.
- **Dataloader**: Implemented batch processing with shuffling for the training set to enhance generalization, a technique validated to improve model robustness in multilingual NLP [6].
- **Procedure**: The model was fine-tuned epoch-wise, with validation performed after each epoch to monitor performance metrics.

5. Results and Discussion

In this study, we experimented with four transformer models like XLM-ROBERTa, IndicBERT, MuRIL, and mBERT with CNN enhancements in different Indian languages for hate speech detection. The models were trained to address four subtasks: Sentiment, Sarcasm, Vulgarity, and Abuse detection. The macro F_1-score evaluation metric was used to evaluate the effectiveness of the model in different subtasks. The macro F_1-score of different transformer models is presented in Table 3. The best performance of a model is shown in bold. The effectiveness of the transformer model was presented graphically in

Figures 1, 2, 3, and 4 for different Indian languages. Among the transformer models evaluated, the MuRIL architecture provided the best effectiveness in different Indian languages. The primary reason was that the MuRIL model was trained in code-mixed Indian languages, improving its ability to capture linguistic nuances. Our observation is similar to that reported by Ghosh et al. [16]. Other transformer models including XLM-ROBERTa, IndicBERT and mBERT also provided competitive performance in different Indian languages. CNN enhancements played a crucial role in improving feature extraction, contributing to high F_1-scores for in different sub-tasks. A high F_1-score demonstrated the effectiveness of the convolutional layer in capturing local patterns within tokenized sequences.

According to the shared task evaluation policy, we have submitted multiple models. Among the models submitted, the MuRIL provided competitive effectiveness in the leaderboard. Our team, ‘CSE_SVNIT’ achieved notable rankings on the HASOC 2025 leaderboard, as shown in Table 4. The macro F_1-score ranking was provided by the organizers, positioning us competitively among the participants and highlighting our ability to handle diverse linguistic contexts. The figures and rankings collectively highlighted our competitive standing within the HASOC 2025 competition, with clear opportunities for improvement in data-scarce scenarios to further elevate performance.

Language	XLM-ROBERTa	IndicBERT	MuRIL	mBERT
Bangla	0.5156	0.5106	**0.5547**	0.5275
Hindi	0.5610	0.5387	**0.5719**	0.5409
Gujarati	0.5808	0.5133	**0.5822**	0.5673
Bodo	0.5257	0.5630	**0.5873**	0.5378

Table 3
Macro F_1-scores of transformer models in different Indian languages

Language	macro F_1-score (Best Model)	Rank/Total Teams
Bangla	0.5547 (MuRIL)	10/17
Hindi	0.5719 (MuRIL)	7/18
Gujarati	0.5822 (MuRIL)	9/15
Bodo	0.5873 (MuRIL)	6/15

Table 4
Ranking of our best performing model in HASOC 2025 shared task

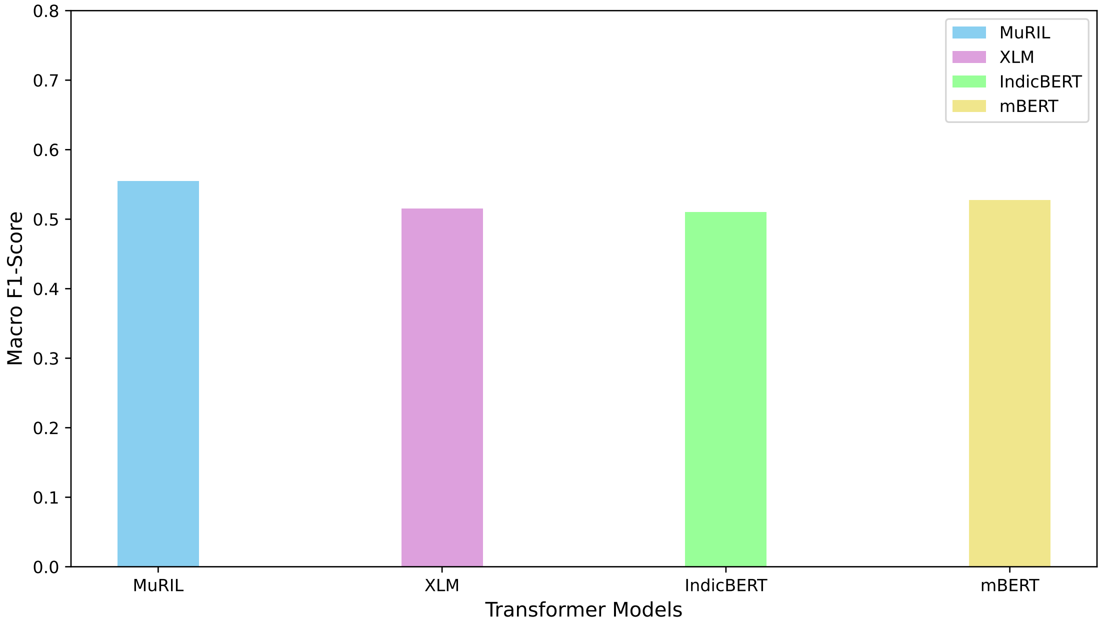

Figure 1: Macro F_1-scores of Transformer models on Bangla

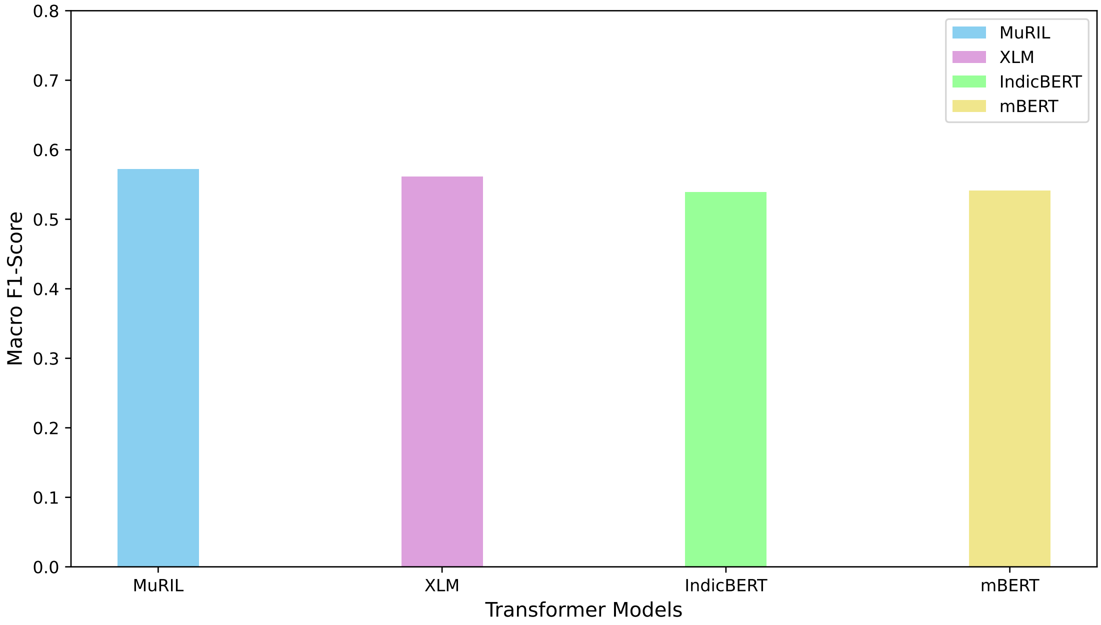

Figure 2: Macro F_1-scores of Transformer models on Hindi

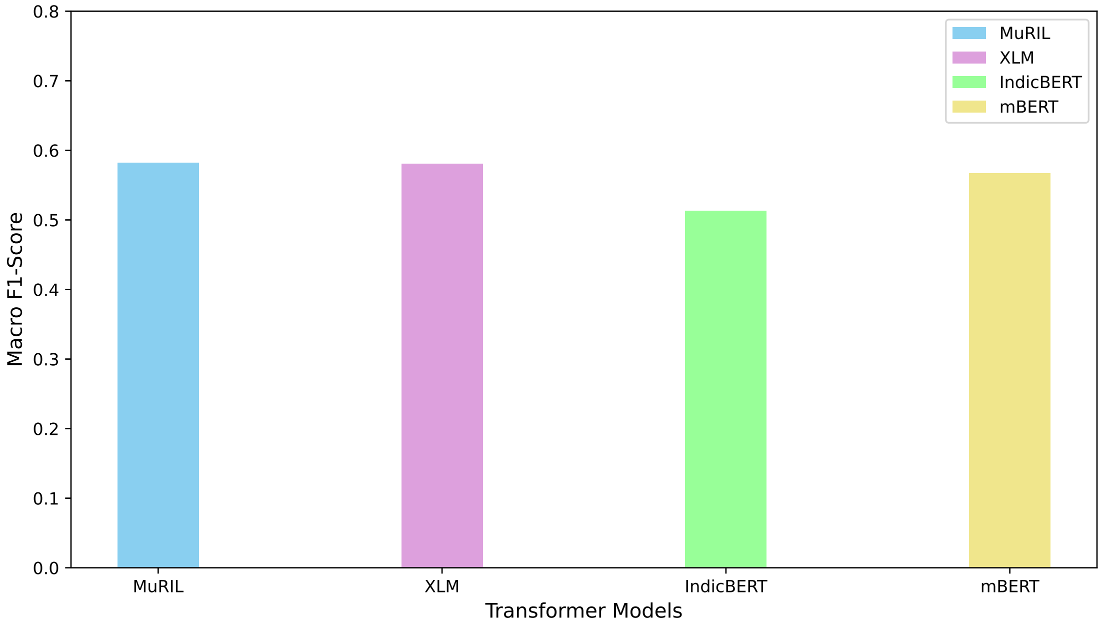

Figure 3: Macro F_1-scores of Transformer models on Gujarati

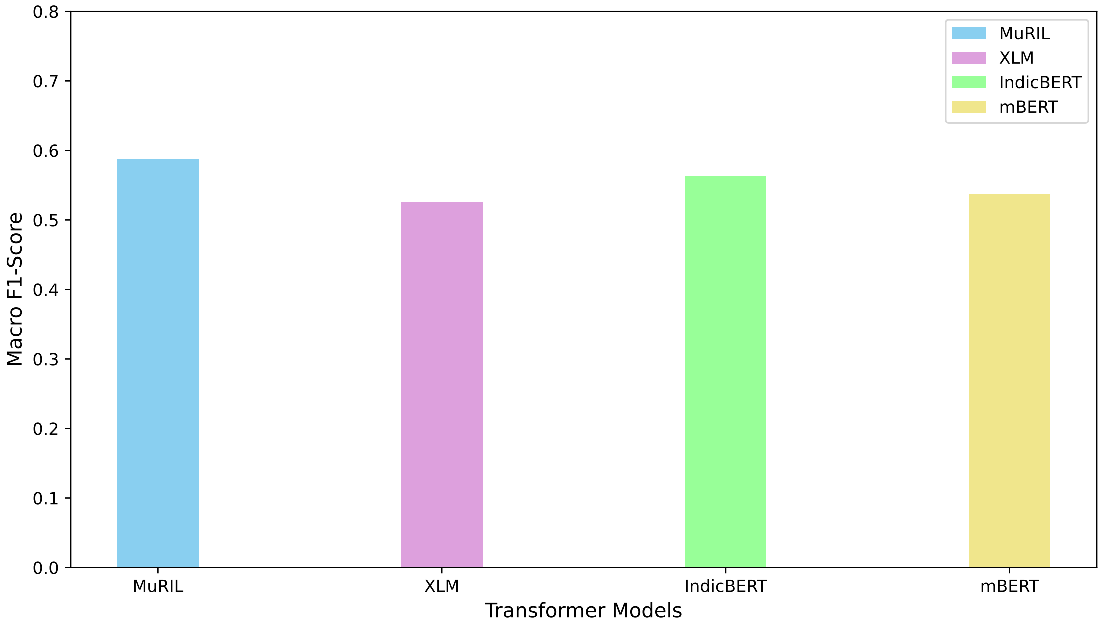

Figure 4: Macro F_1-scores of Transformer models on Bodo

6. Conclusion

Hate speech detection is an important task in the natural language processing domain. In this study, we explore different transformer models including XLM-ROBERTa, IndicBERT, MuRIL, and mBERT, enhanced with layers of convolutional neural network (CNN) layers, for detection of hate speech in different Indian languages. Among the models evaluated, MuRIL provided the best performance and outperforms other models in different Indian languages. Other models like XLM-ROBERTa, IndicBERT, and mBERT also provided competitive performance. Our team, 'CSE_SVNIT' secured notable rankings and provided competitive performance to other submitted models in the HASOC shared task. Despite these successes, the transformer model provided poor performance in the unbalanced data set and the smaller training data. Further fine-tuning of models may improve performance in hate speech detection. These efforts will support the development of more accurate hate speech detection systems, contributing to safer and more inclusive digital spaces across diverse linguistic environments.

Declaration on Generative AI

During the preparation of this article, the author(s) used ChatGPT, QuillBot in order to: Spelling Check, Paraphrase and reword. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[2] M. Bagdouri and D. W. Oard, "On predicting deletions of microblog posts," *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 887–896, 2017.

[3] J. Brownlee, *Deep Learning with Python*. Machine Learning Mastery, 2019.

[4] T. Mandl et al., "Overview of the hasoc track at fire 2019: Hate speech and offensive content identification in indo-european languages," in *Proceedings of the 11th Annual Meeting of the Forum for Information Retrieval Evaluation*, ser. FIRE '19, Kolkata, India: Association for Computing Machinery, 2019, pp. 14–17, ISBN: 9781450377508. DOI: 10.1145/3368567.3368584. [Online]. Available: https://doi.org/10.1145/3368567.3368584.

[5] R. Rajalakshmi and U. S. Reddy, "Feature selection using binary crow search algorithm and its application to text feature selection," *International Journal of Recent Technology and Engineering*, vol. 8, no. 2, pp. 297–304, 2019.

[6] A. Conneau et al., "Unsupervised cross-lingual representation learning at scale," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 8440–8451. DOI: 10.18653/v1/2020.acl-main.747. [Online]. Available: https://aclanthology.org/2020.acl-main.747.

[7] A. K. Das, A. Al Asif, A. Paul, and M. N. Hossain, "Bangla hate speech detection on social media using attention-based recurrent neural network," *Journal of Intelligent Systems*, vol. 30, no. 1, pp. 578–591, 2021.

[8] T. Chakraborty et al., "Multimodal sentiment analysis for low-resource languages using bert," *Journal of Computational Linguistics*, vol. 49, pp. 567–583, 2022.

[9] M. Das, S. Banerjee, and A. Mukherjee, "Data bootstrapping approaches to improve low resource abusive language detection for indic languages," in *Proceedings of the 33rd ACM conference on hypertext and social media*, 2022, pp. 32–42.

[10] A. Gupta et al., “Random forest with TF-IDF features for hate speech detection in HASOC 2022 Bangla dataset,” in *Proceedings of the Forum for Information Retrieval Evaluation (FIRE 2022)*, Reported 83% accuracy on HASOC 2022 Bangla dataset, Kolkata, India: Association for Computing Machinery, Dec. 2022, pp. 1–6. DOI: 10.1145/3574318.3574336. [Online]. Available: https://doi.org/10.1145/3574318.3574336.

[11] D. Nkemelu, H. Shah, M. Best, and I. Essa, “Tackling hate speech in low-resource languages with context experts,” in *Proceedings of the 2022 International Conference on information and communication technologies and development*, 2022, pp. 1–11.

[12] W. A. Abro et al., “Improving hate speech detection with bigram features and svm,” *Journal of Computational Linguistics*, vol. 12, pp. 45–60, 2023.

[13] P. Alonso et al., “Transformer models in hate speech detection: A comparative study,” *Computational Linguistics*, vol. 49, pp. 567–580, 2023.

[14] B. R. Chakravarthi et al., “Offensive language detection in code-mixed dravidian languages,” in *Proceedings of the ACL Workshop on Offensive Language*, 2023.

[15] M. Das and A. Mukherjee, “Banglaabusememe: A dataset for bengali abusive meme classification,” in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023, pp. 15 498–15 512.

[16] A. Ghosh et al., “Mbert and bangla-bert for hate speech in assamese,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, pp. 89–104, 2023.

[17] S. Ghosh and P. Sharma, “Indicbert for sentiment analysis in low-resource indian languages,” *Journal of Artificial Intelligence Research*, vol. 68, pp. 789–805, 2023.

[18] A. Gupta et al., “DistilBERT for offensive language detection in code-mixed Gujarati-English,” *arXiv preprint arXiv:2311.05678*, Nov. 2023, Preprint; focuses on DistilBERT for offensive language in code-mixed Gujarati-English. DOI: 10.48550/arXiv.2311.05678. arXiv: 2311.05678 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2311.05678.

[19] A. Kumar et al., “Enhancing hate speech detection in hindi with trigram features and random forest,” *Proceedings of the 15th Forum for Information Retrieval Evaluation (FIRE)*, 2023.

[20] S. Mutanga et al., “Distilbert for enhanced hate speech detection,” *Natural Language Engineering*, vol. 29, pp. 123–138, 2023.

[21] T. Ranasinghe et al., “Overview of the hasoc subtracks at fire 2023: Hate speech and offensive content identification in assamese, bengali, bodo, gujarati and sinhala,” in *Proceedings of the 15th annual meeting of the forum for information retrieval evaluation*, 2023, pp. 13–15.

[22] S. Rao et al., “Hybrid bidirectional lstm-cnn for hate speech detection in hasoc 2023,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, pp. 101–115, 2023.

[23] P. Singh et al., “SVM with n-grams for multilingual hate speech detection,” in *Proceedings of the 16th Forum for Information Retrieval Evaluation (FIRE 2023)*, Presented at FIRE 2023; focuses on SVM with n-gram features for multilingual hate speech detection in low-resource languages, Goa, India: Association for Computing Machinery, Dec. 2023, pp. 1–6. DOI: 10.1145/3632754.3633277. [Online]. Available: https://doi.org/10.1145/3632754.3633277.

[24] R. Vasudev et al., “Deep learning for hate speech detection on twitter,” *Proceedings of the ACM on Human-Computer Interaction*, vol. 7, pp. 789–800, 2023.

[25] X. Zhang et al., “Convolutional and gated recurrent networks for hate speech detection,” *IEEE Access*, vol. 11, pp. 3456–3467, 2023.

[26] S. Das et al., “A survey on automatic online hate speech detection in low-resource languages,” *arXiv preprint arXiv:2411.19017*, Nov. 2024. DOI: 10.48550/arXiv.2411.19017. arXiv: 2411.19017 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2411.19017.

[27] K. Ghosh, N. K. Singh, J. Mahapatra, et al., “Safespeech: A three-module pipeline for hate intensity mitigation of social media texts in indic languages,” *Social Network Analysis and Mining*, vol. 14, no. 245, 2024. DOI: 10.1007/s13278-024-01393-9. [Online]. Available: https://doi.org/10.1007/s13278-024-01393-9.

[28] A. Gupta et al., “Deep learning frameworks for Hindi hate speech detection on Twitter,” pp. 1–6, Dec. 2024, Focuses on CNN and LSTM models for Hindi hate speech detection on Twitter. DOI: 10.1145/3701091.3701123. [Online]. Available: https://doi.org/10.1145/3701091.3701123.

[29] R. Patel et al., “Roberta for low-resource language classification,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 23, pp. 45–60, 2024.

[30] T. Ranasinghe et al., “Overview of the hasoc subtracks at fire 2023: Hate speech and offensive content identification in assamese, bengali, bodo, gujarati and sinhala,” in *Proceedings of the 15th Annual Meeting of the Forum for Information Retrieval Evaluation*, ser. FIRE ’23, Panjim, India: Association for Computing Machinery, 2024, pp. 13–15, ISBN: 9798400716324. DOI: 10.1145/3632754.3633278. [Online]. Available: https://doi.org/10.1145/3632754.3633278.

[31] A. Singh and R. Thakur, “Generalizable multilingual hate speech detection on low resource indian languages using fair selection in federated learning,” *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics*, vol. 1, 2024.

[32] K. Ghosh et al., “ Overview of the HASOC Track at FIRE 2025: Abusive Meme Identification — Shadows Behind the Laughter,” in *Forum for Information Retrieval Evaluation (Working Notes) (FIRE 2025) December 17-20, Varanasi , India*, K. Ghosh, T. Mandl, and S. Pal, Eds., CEUR-WS.org, 2025.

[33] K. Ghosh et al., “ Overview of the HASOC Track at FIRE 2025: Abusive Meme Identification — Shadows Behind the Laughter,” in *FIRE ’25: Proceedings of the 17th Annual Meeting of the Forum for Information Retrieval Evaluation. December 17-20, Varanasi , India*, New York, NY, USA: Association for Computing Machinery (ACM), 2025.

[34] HASOC 2025 Organizers, *Hate speech and offensive content identification (hasoc) 2025 dataset*, 2025. [Online]. Available: https://hasocfire.github.io/hasoc/2025/.