One_by_zero@NLU of Devanagari Script Languages 2025: Target Identification for Hate Speech Leveraging Transformer-based Approach

Dola Chakraborty, Jawad Hossain and **Mohammed Moshiul Hoque**
Department of Computer Science and Engineering
Chittagong University of Engineering & Technology, Chattogram-4349, Bangladesh
{u1904012, u1704039}@student.cuet.ac.bd, moshiul_240@cuet.ac.bd

Abstract

People often use written words to spread hate aimed at different groups that cannot be practically detected manually. Therefore, developing an automatic system capable of identifying hate speech is crucial. However, creating such a system in a low-resourced languages (LRLs) script like Devanagari becomes challenging. Hence, a shared task has been organized targeting hate speech identification in the Devanagari script. This work proposes a pre-trained transformer-based model to identify the target of hate speech, classifying it as directed toward an individual, organization, or community. We performed extensive experiments, exploring various machine learning (LR, SVM, MNB, GB, and ensemble), deep learning (CNN, LSTM, CNN+BiLSTM), and transformer-based models (IndicBERT, mBERT, MuRIL, XLM-R) to identify hate speech. Experimental results indicate that the IndicBERT model achieved the highest performance among all other models, obtaining a macro F1-score of 0.6785, which placed the team 6th in the task.

1 Introduction

The rapid evolution of social media has revolutionized global communication, enabling users to interact and exchange content instantly. As social media platforms have become central to online interaction, they have also become spaces where hate speech flourishes, often targeting specific individuals, organizations, or communities (Schmid et al., 2024). Addressing hate speech on digital platforms is essential for creating a secure, more inclusive online environment; however, the vast amount of content makes manual detection impractical. This challenge highlights the need for automated hate speech detection systems capable of accurately identifying targets within hateful language. However, hate speech often relies on context and subtle nuances in language, such as sarcasm, humor, or cultural references, making it challenging for automatic systems to identify accurately (Parihar et al., 2021).

In recent years, Natural Language Processing (NLP) has emerged as a promising solution to this problem, with significant advancements in hate speech detection for widely spoken languages (Lemmens et al., 2021). However, identifying specific targets of hate speech in low-resource languages (LRLs), especially those that use scripts like Devanagari (Hindi, Nepali), has received limited attention. The scarcity of resources and high-quality annotated datasets in Devanagari scripts is one of the critical barriers to effective hate speech detection in this script. Devanagari scripts' intricate syntax and semantics often lead to misinterpretations of hate speech, especially in cases involving indirect expressions, ambiguities, cultural allusions, or slang without an understanding of cultural and social context. Addressing these gaps, a shared task (Thapa et al., 2025) is organized at CHIPSAL@COLING2025 (Sarveswaran et al., 2025) that focuses on identifying the specific targets of hate speech within the Devanagari-script text. In this task, each instance of hate speech is categorized by its intended target, an individual, organization, or community, to deepen understanding of the scope and direction of hateful expressions in these languages. As participants in the task, the critical contributions of our work are outlined below.

- Developed a transformer-based model to categorize hate speech by its intended target: individual, organization, or community.
- Examined various baselines, including machine learning (ML), deep learning (DL), and transformers to perform the tasks.

2 Related Work

A wide range of studies have been conducted in NLP regarding hate speech. Dhanya and Balakrishnan (2021) explored the detection of hate speech

Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025), pages 327–333
January 19, 2025.

in various Asian languages, focusing on developing an automated system tailored for Malayalam. Shvets et al. (2021) worked on identifying sexism and racism in social media posts. Using GetTA Pair with BERT resulted in lower accuracy with 0.57 on the test set for exact matches but achieved a considerably higher accuracy of 0.82 for partial matches. The challenge of detecting hatred and insulting language in an LRL (Telugu), which is also code-mixed, was addressed by Farsi et al. (2024). They employed sentence BERT, achieving a macro F1-score of 0.70. Plaza-del Arco et al. (2021) used multi-task learning with sentiment, emotion, and target detection to recognize hate and offensive language. They implemented a multi-head, multi-task learning model based on BERT, which achieved the highest F1 score of 0.862. Farooqi et al. (2021) addressed hate speech detection on social media tweets, comments, and replies. They used code-mixed data (Hindi + English), and their system achieved a macro F1-score of 0.72 leveraging neural networks and ensemble transformer-based models (IndicBERT, XLM-ROBERTA, Multilingual BERT). A study by Joshi and Joshi (2023) assessed the effectiveness of several sentence-BERT models, including Bengali-SBERT, Gujarati-SBERT, Assamese-BERT, and L3Cube Indic-SBERT, which demonstrated state-of-the-art performance in detecting hate speech across Indian languages. Alam et al. (2024) conducted hate speech detection in Tamil on social media, specifically targeting caste and migration status. They explored various ML, DL, and transformer-based models, including M-BERT, XLM-R, and Tamil BERT. Notably, the M-BERT model achieved a standout macro F1-score of 0.80, marking the highest performance among the models tested. Mossie and Wang (2020) conducted vulnerable community identification on social media posts and comments in both Amharic and English text. The RNN-GRU model outperformed others, achieving an accuracy of 0.92. Singh et al. (2023) implemented the XLM-Roberta-base model on multilingual text data from the 'CrisisHateMM' dataset related to the Russia-Ukraine conflict, achieved the highest performance in detecting hate speech and identifying targets (individual, community, organization) across both Sub-task 1 (text-embedded image hate speech detection) and Sub-task 2 (target detection), with F1 scores of 84.62 and 69.73, respectively. Another notable work emphasizes hate speech detection in Marathi by Velankar et al. (2022) using the L3Cube-MahaHate dataset with 25,000 tweets where monolingual models like MahaBERT (0.909 accuracy for binary) and MahaRoBERTa (0.903 for 4-class) outperformed multilingual BERT variants. Karim et al. (2021) conducted hate speech detection in Bangla, using 8,087 labeled examples from Facebook, YouTube comments, and newspapers, and achieved an 88% F1-score with DeepHateExplainer, an ensemble of Bangla BERT-base, mBERT, and XLM-RoBERTa.

Numerous studies focus on identifying hate speech but lack target-specific classifications, especially for Nepali tweets. There is a vacuum in target identification in Nepali-language detection since most research has been on code-mixed Hindi-English or only Hindi scripts. This work addresses the gaps by including Hindi and Nepali tweets in the Devanagari script. Focusing on target identification in the Devanagari scripts, this work incorporates culturally relevant patterns to enhance the detection of nuanced hate speech.

3 Task and Dataset Description

In the shared task[1], we focus on identifying specific targets within hate speech written in the Devanagari script (Thapa et al., 2025). The task aims to classify each instance of hate speech according to its intended target: *Individual (InD)*, *Organization (OrG)*, and *Community (CoM)*. According to (Jafri et al., 2024), the definition of class is defined as:

- **Individual (InD):** Refers to hateful acts towards specific individuals, such as a self-reliant person targeted.
- **Organization (OrG):** Denotes hate targeted to institutions or groups of people formed to achieve specific goals.
- **Community (CoM):** Indicates instances where hate speech targets communities or larger socioeconomic groups.

The dataset (Jafri et al., 2024, 2023; Thapa et al., 2023; Rauniyar et al., 2023; Ojha, 2019; Kulkarni et al., 2021; Aralikatte et al., 2021) is developed for identifying the target of hate speech, comprises a variety of social media tweets containing hate speech directed toward individuals, organizations, and communities. This task aims to detect and

[1] https://github.com/therealthapa/chipsal24

prevent hate speech directed at individuals, organizations, and communities. The primary goal is to foster a safer and more respectful social media environment. Appendix A provides statistical details of the dataset, outlining key metrics and distributions.

4 Methodology

Figure 1 shows a schematic process in detecting hate speech, illustrating each major phase.

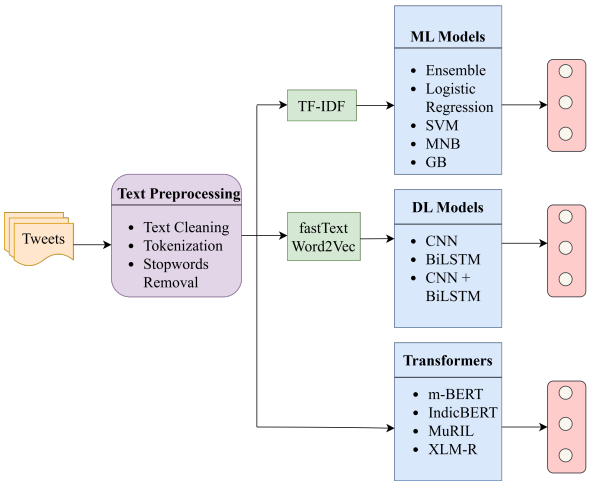

Figure 1: Schematic process of the target identification for hate speech.

4.1 Data Pre-processing

The dataset comprises tweets with substantial unnecessary and redundant content. The tweets contain emojis, unwanted spaces, symbols, punctuations, and URLs. To enhance data quality and handle class imbalance, we implemented a robust preprocessing pipeline that involves text preprocessing and oversampling. The dataset is refined by removing emojis, extraneous symbols, unnecessary punctuation, and URLs that do not significantly aid in identifying the target class. The tweets are then tokenized, with Hindi and Nepali stopwords systematically removed, resulting in a dataset containing only meaningful and relevant information. Appedix B presents the statistics of the training dataset after oversampling, highlighting key changes in data distribution.

4.2 Feature Extraction

We employed distinct feature extraction techniques for ML and DL models, optimizing each approach for its specific strengths in text data comprehension. To optimize performance, the feature set is restricted to the top 5000 terms, balancing the need for interpretability and computational efficiency. Word2Vec and FastText embeddings are employed for DL models, with each embedding represented in a 300-dimensional vector space. These embeddings capture semantic relationships between words, crucial for understanding context in the nuanced language of hate speech. Word2Vec offers continuous representations, while FastText incorporates subword information, making it particularly effective for processing morphologically complex languages like Hindi and Nepali.

4.3 ML Models

Various machine learning models are leveraged for target identification of hate speech. Logistic Regression (LR), Support Vector Machine (SVM), Multinomial Naive Bayes (MNB), and Gradient Boosting Classifier (GBC) were implemented. LR is set with a maximum iteration of 1000 for convergence. We also performed hyperparameter tuning for the SVM using "GridSearchCV" to identify the optimal configuration. An ensemble model using a Voting Classifier combining LR, SVM, MNB, and GBC with soft voting was used to enhance classification accuracy. Each model is configured to optimize performance: LR with a maximum iteration of 1000 for convergence, SVM with probability enabled for soft voting, MNB operates on TF-IDF vectorized term frequencies, and GBC refines predictions on complex samples. This ensemble leverages the strengths of each model to improve target identification.

4.4 DL Models

Three DL models, CNN, BiLSTM, and CNN+BiLSTM, were employed for the task. The CNN model utilizes an embedding layer with pre-trained word vectors, followed by a 1D convolutional layer with 128 filters and a kernel size of 5, along with max pooling and global max pooling layers to extract features and reduce dimensionality. It includes a dense layer with 64 neurons and a dropout rate of 0.5, culminating in a sigmoid activation output for binary classification. The BiLSTM model also begins with an embedding layer and employs a bidirectional LSTM layer with 64 units, leveraging forward and backward context. This is followed by global max pooling and a dense layer structure with dropout. The CNN-BiLSTM model integrates both architectures, featuring a convolutional layer for local pattern extraction and a bidirectional

LSTM for contextual understanding. All these three models are compiled with the 'Adam' optimizer and trained using 'binary_crossentropy' loss for 5 epochs.

4.5 Transformers

We utilized various pre-trained transformer-based models from HuggingFace[2] to identify the one that performs best for our task. We employed transformer-based models such as m-BERT, IndicBERT, MuRIL, and XLM-R. IndicBERT outperformed all other ML, DL, and transformer-based models by achieving the highest macro F1 score.

IndicBERT is a pre-trained multilingual language model designed to process multiple Indic languages and English. It is trained on the IndicCorp v2 dataset and evaluated against the IndicXTREME benchmark, showcasing its robustness in understanding diverse linguistic contexts. With a parameter count of 278 million, the model supports 23 Indic languages, enhancing its versatility in natural language processing tasks. The fine-tuned model architecture comprises a pre-trained IndicBERT with three output labels. IndicBERT-MLM utilizes a vanilla BERT architecture trained with the Masked Language Model.

Table 1 demonstrates the hyperparameters that are fine-tuned to attain best performance of the IndicBERT model.

Hyperparameter	Value
Optimizer	AdamW
Learning Rate	2e-5
Batch Size	16
Max Length	128
Weight Decay	0.05
Epochs	3

Table 1: Hyperparameter setup for transformer-based models

The hyperparameters are fine-tuned through extensive experimentation. Various learning rates, including 1e-5 and 5e-5, are tested, along with different batch sizes such as 4, 8, and 32, to evaluate performance. The maximum sequence length is set to 128 for optimal model generalization. Multiple epoch configurations, such as 5, 10, and 15, are implemented. After thorough trials, the model performs best with these selected hyperparameters.

[2] https://huggingface.co/

5 Results and Discussion

Table 2 illustrates the performance of the various ML, DL, and transformer-based models employed on the test set. The model's performance is evaluated using the macro F1-score. Among ML models, the LR model achieved the highest macro F1-score of 0.5267, while the ensemble model closely follows with a score of 0.5220. The ensemble model highlights potential challenges in integrating diverse ML models such as LR, MNB, and GB. The other ML models had an F1 score slightly lower than this value. DL models exhibited inferior performance compared to ML models. The CNN model with FastText embeddings yields a lower macro F1-score of 0.2175, while the BiLSTM model achieved a macro F1-score of 0.4587. The ensemble of CNN and BiLSTM achieved a higher F1-score of 0.5046. 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.

Model	P	R	F1
Ensemble	0.52	0.52	0.52
LR	0.53	0.53	0.53
SVM	0.46	0.47	0.46
MNB	0.51	0.52	0.51
GB	0.48	0.47	0.47
CNN	0.16	0.33	0.22
BiLSTM	0.46	0.46	0.46
CNN + BiLSTM	0.50	0.51	0.50
m-BERT	0.59	0.58	0.58
IndicBERT	**0.69**	**0.67**	**0.68**
MuRIL	0.68	0.68	0.68
XLM-R	0.63	0.63	0.63

Table 2: Performance of various ML, DL, Transformer-based models on the test set. P (Precision), R (Recall), F1 (macro F1-score)

The results highlight the superiority of transformer-based models in handling linguistic diversity while considering the limitations of conventional DL approaches, particularly in capturing the rich semantic information for LRLs needed for accurate classification. Lack of pre-trained models in ML/DL specially tailored for LRLs can be a key reason for such poor performence. A detailed error analysis of the best-performed model, both

quantitative and qualitative, is presented below to offer a comprehensive understanding of the proposed model's performance.

Quantitative Analysis: An in-depth quantitative error analysis is done using the confusion matrix (Figure 2). The confusion matrix depicts that a total of 345 samples are classified correctly out of 475 samples. A total of 34 samples from the *InD* class are misclassified as *OrG*, while 18 are mistaken for *CoM*. Similarly, 34 samples from the *OrG* class are misclassified as *InD*, with an additional 13 misclassified as *CoM*. Meanwhile, 19 samples from the *CoM* class are misclassified as *InD*, and 12 as *OrG*. The misclassification can be traced to the initial class imbalance in the dataset. Though oversampling is performed, new feature patterns are not integrated, leading to a bias toward the limited features. This residual bias potentially impacts its ability to generalize effectively across all classes.

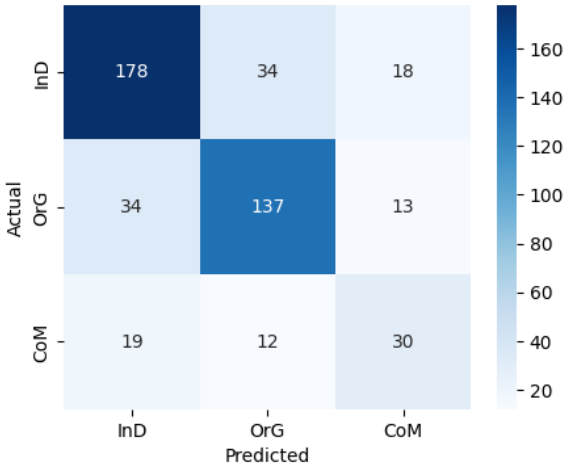

Figure 2: Confusion matrix of the best-performed model (IndicBERT).

Qualitative Analysis: A comparison of actual labels and predicted labels on a particular tweet is illustrated in Figure 3. The first three samples are predicted correctly as their actual classes. However, the fourth sample is incorrectly predicted. It is meant to target a community, but our model wrongly predicts the sample as *InD*.

The misclassifications likely occur due to the inherent challenges in context understanding and the overlap of semantic features between the classes. Even with IndicBERT, which excels in multilingual tasks, subtle contextual cues in the Devanagari script may cause confusion between classes.

Tweets	True Label	Predicted Label
केही बर्षमा कांग्रेस भित्र माओबादी मात्रै बाकी रलान जस्तो छ, राजीनामा राजीनामा, भनष्कासन गनष थाले कांग्रेभस	OrG	OrG
कस्ता पार्टी सभापति हुन्, हँसियाहथौडामा भाेट नहाले कारबाही रे । मारेके	InD	InD
नेपाली जनता श्रीलंकाको जनता बन्न समय लाग्छ। नेपाली जनताले दुख भोग्न पुगेको छैन।	CoM	CoM
सबै मिली सत्रे लाई हराऔँ	CoM	InD

Figure 3: Some predicted outputs by the IndicBERT.

6 Conclusion

This study evaluates several ML, DL, and transformer-based models for identifying hate speech targets in Hindi and Nepali tweets written in Devanagari script. While traditional ML methods like LR and ensemble provided valuable insights, they struggled to capture complex semantic relationships. DL models also faced challenges with feature representation in the Devanagari script and obtained poor results. However, IndicBERT outperformed all other ML and DL approaches among transformer-based models, achieving the highest F1-score of 0.6785 by effectively capturing the nuances of the Devanagari script. Future work can explore advanced embeddings, hybrid models, or ensemble multiple transformers for enhanced performance in hate speech detection.

7 Limitations

The current work poses several constraints: (i) The presented method relies on the pre-trained IndicBERT, which may require further fine-tuning and modification to capture contextual patterns better. (ii) The dataset is imbalanced, and to address this, we applied the oversampling technique, resampling the minority class. However, this approach may limit the model's ability to learn diverse patterns, impacting its performance. New NLP augmentation techniques can be more helpful in further investigation. (iii) DL models' performance can be investigated further, exploring alternative embeddings and classifiers.

References

Md Alam, Hasan Mesbaul Ali Taher, Jawad Hossain, Shawly Ahsan, and Mohammed Moshiul Hoque. 2024. Cuet_nlp_manning@ lt-edi 2024: Transformer-based approach on caste and migration hate speech detection. In *Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion*, pages 238–243.

Rahul Aralikatte, Miryam De Lhoneux, Anoop Kunchukuttan, and Anders Søgaard. 2021. Itihasa: A large-scale corpus for sanskrit to english translation. In *Proceedings of the 8th Workshop on Asian Translation (WAT2021)*, pages 191–197.

L. K. Dhanya and Kannan Balakrishnan. 2021. Hate speech detection in asian languages: a survey. In *2021 International Conference on Communication, Control and Information Sciences (ICCISc)*, volume 1. IEEE.

Zaki Mustafa Farooqi, Sreyan Ghosh, and Rajiv Ratn Shah. 2021. Leveraging transformers for hate speech detection in conversational code-mixed tweets. *arXiv preprint arXiv:2112.09986*.

Salman Farsi, Asrarul Eusha, Jawad Hossain, Shawly Ahsan, Avishek Das, and Mohammed Moshiul Hoque. 2024. Cuet_binary_hackers@ dravidianlangtech eacl2024: Hate and offensive language detection in telugu code-mixed text using sentence similarity bert. In *Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 193–199.

Farhan Ahmad Jafri, Kritesh Rauniyar, Surendrabikram Thapa, Mohammad Aman Siddiqui, Matloob Khushi, and Usman Naseem. 2024. Chunav: Analyzing hindi hate speech and targeted groups in indian election discourse. *ACM Transactions on Asian and Low-Resource Language Information Processing*.

Farhan Ahmad Jafri, Mohammad Aman Siddiqui, Surendrabikram Thapa, Kritesh Rauniyar, Usman Naseem, and Imran Razzak. 2023. Uncovering political hate speech during indian election campaign: A new low-resource dataset and baselines.

Ananya Joshi and Raviraj Joshi. 2023. Harnessing pretrained sentence transformers for offensive language detection in indian languages. *arXiv preprint arXiv:2310.02249*.

Md. Rezaul Karim, Sumon Kanti Dey, Tanhim Islam, Sagor Sarker, Mehadi Hasan Menon, Kabir Hossain, Md. Azam Hossain, and Stefan Decker. 2021. Deephateexplainer: Explainable hate speech detection in under-resourced bengali language. In *2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 1–10.

Atharva Kulkarni, Meet Mandhane, Manali Likhitkar, Gayatri Kshirsagar, and Raviraj Joshi. 2021. L3cubemahasent: A marathi tweet-based sentiment analysis dataset. In *Proceedings of the Eleventh Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 213–220.

Jens Lemmens, Ilia Markov, and Walter Daelemans. 2021. Improving hate speech type and target detection with hateful metaphor features. In *Proceedings of the Fourth Workshop on NLP for Internet Freedom: Censorship, Disinformation, and Propaganda*.

Zewdie Mossie and Jenq-Haur Wang. 2020. Vulnerable community identification using hate speech detection on social media. *Information Processing & Management*, 57(3):102087.

Atul Kr Ojha. 2019. English-bhojpuri smt system: Insights from the karaka model. *arXiv preprint arXiv:1905.02239*.

Anil Singh Parihar, Surendrabikram Thapa, and Sushruti Mishra. 2021. Hate speech detection using natural language processing: Applications and challenges. In *2021 5th International Conference on Trends in Electronics and Informatics (ICOEI)*, pages 1302–1308. IEEE.

Flor Miriam Plaza-del Arco, Sercan Halat, Sebastian Padó, and Roman Klinger. 2021. Multi-task learning with sentiment, emotion, and target detection to recognize hate speech and offensive language. *arXiv preprint arXiv:2109.10255*.

Kritesh Rauniyar, Sweta Poudel, Shuvam Shiwakoti, Surendrabikram Thapa, Junaid Rashid, Jungeun Kim, Muhammad Imran, and Usman Naseem. 2023. Multi-aspect annotation and analysis of nepali tweets on anti-establishment election discourse. *IEEE Access*.

Kengatharaiyer Sarveswaran, Bal Krishna Bal, Surendrabikram Thapa, Ashwini Vaidya, and Sana Shams. 2025. A brief overview of the first workshop on challenges in processing south asian languages (chipsal). In *Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL)*.

Ursula Kristin Schmid, Anna Sophie Kümpel, and Diana Rieger. 2024. *How social media users perceive different forms of online hate speech: A qualitative multi-method study*, volume 26.

Alexander Shvets et al. 2021. Targets and aspects in social media hate speech. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*.

Karanpreet Singh, Vajratiya Vajrobol, and Nitisha Aggarwal. 2023. Iic_team@ multimodal hate speech event detection 2023: Detection of hate speech and targets using xlm-roberta-base. In *Proceedings of the 6th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text*.

Surendrabikram Thapa, Kritesh Rauniyar, Farhan Ahmad Jafri, Surabhi Adhikari, Kengatharaiyer Sarveswaran, Bal Krishna Bal, Hariram Veeramani, and Usman Naseem. 2025. Natural language understanding of devanagari script languages: Language identification, hate speech and its target detection. In *Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL)*.

Surendrabikram Thapa, Kritesh Rauniyar, Shuvam Shiwakoti, Sweta Poudel, Usman Naseem, and Mehwish Nasim. 2023. Nehate: Large-scale annotated data shedding light on hate speech in nepali local election

discourse. In *ECAI 2023*, pages 2346–2353. IOS Press.

Abhishek Velankar, Hrushikesh Patil, Amol Gore, Shubham Salunke, and Raviraj Joshi. 2022. L3cube-mahahate: A tweet-based marathi hate speech detection dataset and bert models. *arXiv preprint arXiv:2203.13778*.

A Appendix

Classes	Train	Valid	Test	T_W	T_{UW}
InD	1074	230	230	42438	15681
OrG	856	183	184	31181	11722
CoM	284	61	61	10564	5182
Total	2214	474	475	84183	25731

Table A.1: Dataset Statistics for Train, Validation, and Test Sets. T_W and T_{UW} denotes total words and total unique words, respectively

Table A.1 demonstrates the statistics of the dataset. The dataset comprises a total of 3163 instances of hate speech. The training dataset consists of 2214 samples. In addition, the validation dataset includes 474 tweets, while the test set contains 475 tweets, which will be used in the final evaluation to assess the model's generalization to unseen data. The datasets are highly imbalanced, with the individual class containing substantially more instances than the organization and community classes.

B Appendix

Analyzing the dataset reveals a substantial class imbalance, with instances of the class *InD* being the most prevalent, while *OrG* and *CoM* classes are significantly underrepresented. To address this, we employed an oversampling technique specifically targeting the minority classes. We accomplished this by replicating samples from these underrepresented classes until the number of instances in each class was comparable. This approach ensures that the model obtains sufficient samples from each class, minimizing the risk of bias toward the majority class. Table B.1 shows the number of training instances after applying the oversampling technique.

Classes	Train	T_W	T_{UW}
Individual	1074	29471	11963
Organization	1074	27831	6931
Communication	1074	27886	3773
Total	3222	85188	18187

Table B.1: Statistics of training dataset after oversampling

Oversampling can lead to bias by artificially inflating the representation of minority classes. Since it just duplicates existing minority class examples instead of generating truly novel samples, the new observations do not provide additional informative details about under-represented classes. This reduces the model's ability to generalize to unseen data and increases the risk of overfitting. To overcome this problem new data augmentation techniques introduced in NLP can be used in further analysis for better results. Methods like back translation, synonym replacement, lexical substitution, noise injection can enhance linguistic diversity and make models robust to minor changes. For target identification in hate speech in the Devanagari script, context-aware substitution and adversarial methods can help to reduce bias and improve generalization.