Paramananda@NLU of Devanagari Script Languages 2025: Detection of Language, Hate Speech and Targets using FastText and BERT

Darwin Acharya
Kathmandu University
acharyadarwin5@gmail.com

Sundeep Dawadi
Kathmandu University
sundeepdwd@gmail.com

Shivram Saud
Kathmandu University
saudshivram373@gmail.com

Sunil Regmi
Kathmandu University
sunil.regmi@ku.edu.np

Abstract

This paper presents a comparative analysis of FastText and BERT-based approaches for Natural Language Understanding (NLU) tasks in Devanagari script languages. We evaluate these models on three critical tasks: language identification, hate speech detection, and target identification across five languages: Nepali, Marathi, Sanskrit, Bhojpuri, and Hindi. Our experiments, although with a raw tweet dataset but extracting only the Devanagari script, demonstrate that while both models achieve exceptional performance in language identification (F1 scores > 0.99), they show varying effectiveness in hate speech detection and target identification tasks. FastText with augmented data outperforms BERT in hate speech detection (F1 score: 0.8552 vs. 0.5763), while BERT shows superior performance in target identification (F1 score: 0.5785 vs. 0.4898). These findings contribute to the growing body of research on NLU for low-resource languages and provide insights into model selection for specific tasks in Devanagari script processing.

1 Introduction

The proliferation of digital content in Devanagari script languages has created an urgent need for robust Natural Language Understanding (NLU) tools (Wilie et al., 2020). These tools are essential for content moderation, ensuring safe online spaces, and preserving linguistic diversity in digital platforms (Parihar et al., 2021; Kumar et al., 2024). However, processing Devanagari script languages presents unique challenges due to their complex character sets, morphological richness, and limited computational resources (Sarveswaran et al., 2025; Thapa et al., 2025).

This research addresses these challenges through two primary approaches. Firstly, the implementation of FastText (Joulin et al., 2017), known for its efficiency in handling Devanagari text data (Bansod, 2023; GitHub), and secondly, the utilization of pre-trained BERT-based Multilingual Cased models (Devlin et al., 2019) fine-tuned for specific tasks.

Our work focuses on providing a comprehensive comparison of traditional and transformer-based approaches, establishing baseline performances for three crucial NLU tasks, and identifying optimal model configurations for FastText Devanagari script processing tasks.

2 Related Work

Recent research in Devanagari script processing has focused on developing robust language identification systems and hate speech detection mechanisms (Kumbhar and Thakre, 2024; Rauniyar et al., 2023). Language identification and hate speech detection in low-resource languages, particularly those using the Devanagari script, have garnered significant research attention due to the increasing digital content in these languages. Traditional methods for language identification often relied on statistical models and n-gram analyses. With the advent of deep learning, more sophisticated models have emerged, offering improved performance (Bansod, 2023).

In the context of Indic languages, AI4Bharat's IndicLID (Devlin et al., 2019) leveraged FastText-based models fine-tuned on multiple Indian languages for language identification. Their models demonstrated high precision, recall, and F1-scores, with significant throughput suitable for large-scale applications. For instance, the IndicLID-FTN-4-dim model achieved an F1-score of 0.99 and an accuracy of 0.98, outperforming models like CLD3 and NLLB in terms of throughput and model size.

Thapa et al. (Thapa et al., 2023a) conducted the Multimodal Hate Speech Event Detection shared task at CASE 2023, providing valuable insights into various methodologies for hate speech detection. The methods from different participants re-

Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025), pages 334–338
January 19, 2025. ©2025 International Committee on Computational Linguistics

vealed interesting approaches, with transformer-based methods proving to be more effective. Most participants utilized BERT-based variations to extract textual features from the dataset (Hürriyetoğlu et al., 2023).

Bansod (Bansod, 2023) explored hate speech detection in Hindi using various embedding methods, including FastText, GloVe, and transformer-based embeddings like DistilBERT and MuRIL. The study found that transformer-based models, particularly when fine-tuned with low learning rates and class weights, achieved macro F1-scores in the range of 70–75%. The research highlighted challenges such as the model's difficulty in detecting sarcasm, understanding veiled references, and the need for background knowledge to interpret certain types of hate speech.

These studies underscore the importance of model selection, data augmentation, and handling linguistic nuances in low-resource languages. They highlight the challenges posed by unbalanced datasets, code-mixed languages, sarcasm, and the necessity for comprehensive datasets that capture the diversity of language use on online platforms. The collective findings contribute to the growing body of research on natural language understanding for Devanagari script languages and provide insights into optimal model configurations for specific tasks in this domain.

The theoretical foundations of our approach build upon the FastText architecture introduced by Joulin et al. (Joulin et al., 2017) and enhanced by Bojanowski et al. (Bojanowski et al., 2017) with subword information. The BERT-based component utilizes the multilingual model developed by Devlin et al. (Devlin et al., 2019), which has shown remarkable effectiveness in cross-lingual tasks.

3 Methodology

In this section, we outline our methodology in a step-by-step manner.

3.1 Task and Dataset Description

The shared task comprised of three specific subtasks: Sub-Task A involved classifying text into five distinct Devanagari languages. Sub-Task B focused on the binary classification challenge of determining whether a given text contained hate speech or not. Sub-Task C focused on identifying the target of hate speech.

3.1.1 Sub-Task A: Language Identification

This problem involved classifying text into five distinct Devanagari languages- Nepali, Marathi, Sanskrit, Bhojpuri, and Hindi - labeled as 0 through 4. The dataset comprised a total of 52,422 training samples, 11,233 evaluation data, and 11,234 test data.

3.1.2 Sub-Task B: Hate Speech Detection

Sub-task B is focused on binary classification of hate speech labeled as 0 ('non-hate') and 1 ('hate'). The dataset comprised a total of 19,019 training samples of text, 4,076 evaluation data, and 4,076 test data.

3.1.3 Sub-Task C: Target Identification

Sub-Task C focused on identifying the targets of hate speech i.e., for whom the hate speech was delivered. The dataset for this sub-task comprised of a total of 2,214 training samples, 474 evaluation data, and 475 test data. There are three classes in the dataset 'individual', 'organization', and 'community' labeled as 0, 1, and 2 respectively.

Dataset Description: Our study utilizes a comprehensive dataset (CodaLab), comprising sentences in five Devanagari script languages. The dataset incorporates diverse sources, including the CHUNAV dataset for Hindi hate speech (Jafri et al., 2024), the Political Hate Speech Corpus (Jafri et al., 2023), the Nehate Nepali hate speech dataset (Thapa et al., 2023b), the Multi-aspect Nepali tweet corpus (Rauniyar et al., 2023), the English-Bhojpuri parallel corpus (Ojha, 2019), the L3CubeMahaSent Marathi dataset (Kulkarni et al., 2021), and the Itihasa Sanskrit-English corpus (Aralikatte et al., 2021).

3.2 Preprocessing of Data

Initially, we cleaned the provided dataset into three sets: training, evaluation, and testing. Using a custom approach defined manually with the Python regular expression library, we extracted only the Devanagari text from the dataset, completely ignoring URLs, emojis, hashtags, mentions, digits, and punctuation, as they were considered irrelevant to the classification problem.

3.3 Data Augmentation

To balance the instances of the 'hate' class in Sub-Task B, the samples from Sub-Task C were merged with Sub-Task B and labeled as 'hate' (label 1).

This was feasible because all the samples in Sub-Task C represented hate tweets but targeted different groups so after augmenting the data we removed the duplicates.

3.4 Model Architecture and Training

Because the deep learning model can learn the complex distribution characteristics of data through deep artificial neural networks and nonlinearity, especially the use of deep learning in tasks related to text data has attracted more and more attention (Zhang et al., 2018).

3.4.1 FastText Implementation

FastText is a library for efficient learning of word representations and sentence classification and obtains performance on par with recently proposed methods inspired by deep learning while being much faster (FastText; Joulin et al., 2017) . It requires minimal preprocessing to preserve linguistic nuances. We have trained the fastText model for Sub-Task A, B, and C. The Hyperparameters we set were Epochs: 500 Learning rate: 1 Embedding dimension: 100 Word N-gram: 1 Bucket size: 10,000. For Task B, we implemented data augmentation strategies for FastText to assess the impact of additional training data, following methodologies validated in recent studies (Bansod, 2023; GitHub). This highly improved the performance of the model which is later discussed in the result section.

3.4.2 BERT Implementation

BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019). All three sub-tasks were fine-tuned on BERT Base Multilingual Uncased and BERT Base Uncased with a constant learning rate of 1e-4 and a batch size of 32, while the number of epochs was varied across 5, 10, and 20.

4 Results and Analysis

This section is dedicated to a comparative detailed analysis of the proposed models on all three sub-tasks. We conducted controlled experiments for each task, maintaining consistent evaluation metrics across models. The performance metrics in Table 1 show the F1 score of each model and provide insight into their effectiveness.

Task	Method	F1 Score (Test)
Language Identification	FastText	0.9917
	BERT	**0.9939**
Hate Speech Detection	FastText	0.6159
	FastText (aug)	**0.8552**
	BERT	0.5763
Target Identification	FastText	0.4898
	BERT	**0.5785**

Table 1: F1 Scores for different NLU Tasks in Devanagari Script

4.1 Quantitative Results

4.2 Analysis

The results reveal distinct model strengths across various Devanagari language processing tasks:

1. **Language Identification:** Both FastText and BERT performed exceptionally well, achieving near-perfect F1 scores (0.9917 and 0.9939, respectively). These results align with previous findings in Indic language processing (GitHub), demonstrating that both models effectively differentiate between Nepali, Marathi, Sanskrit, Bhojpuri, and Hindi.

2. **Hate Speech Detection:** The performance of the models diverged significantly. FastText, when combined with data augmentation, achieved a notable improvement in F1 score from 0.6159 to 0.8552, outperforming BERT substantially. BERT, despite its capacity for deep contextual understanding, struggled with this task, displaying an F1 score of only 0.5763. This underperformance, coupled with signs of overfitting (an evaluation score of 0.88 but a lower test score), indicates that BERT's generalization ability is limited when faced with sparse hate speech datasets.

3. **Target Identification:** For this more nuanced task, BERT outperformed FastText, with F1 scores of 0.5785 versus 0.4898, respectively. This suggests that BERT's contextual embeddings are better suited to identifying and distinguishing complex targets, such as individuals, organizations, and communities, within text. Tuning FastText hyperparameters yielded only minor improvements (±2%), emphasizing its robustness but also its limitations in handling contextual nuances.

5 Conclusion and Future Work

This study provides an in-depth comparative analysis of FastText and BERT models for processing Devanagari script languages. Key findings include:

1. **LanguageIdentification:** Both models excel in distinguishing among Devanagari languages, indicating their robustness in handling script-based variations.

2. **Hate Speech Detection:** FastText, particularly when augmented with additional data, outperforms BERT, highlighting the importance of data volume and diversity. BERT's tendency to overfit suggests a need for more rigorous fine-tuning, especially for low-resource hate speech datasets.

3. **Target Identification:** BERT's superior performance in this task underscores the advantage of leveraging contextual embeddings to capture subtle, nuanced relationships.

Future Directions:

Exploring hybrid approaches that integrate the strengths of both models FastText and BERT could improve overall performance. Investigating script-specific pre-processing techniques to enhance model accuracy. Applying transfer learning techniques to better adapt models to low-resource Devanagari languages, could potentially reduce the need for extensive data augmentation.

6 Acknowledgments

We thank the organizers of CHiPSAL@COLING 2025 (Sarveswaran et al., 2025; Thapa et al., 2025) for providing the datasets and their support throughout this research.

References

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.

Pranjali Prakash Bansod. 2023. Hate Speech Detection in Hindi. Master's thesis, Master's Projects.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

CodaLab. CHIPSAL@COLING 2025 Competition. https://codalab.lisn.upsaclay.fr/competitions/20000#participate-get_data.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

FastText. Supervised-tutorial. https://fasttext.cc/docs/en/supervised-tutorial.html. (accessed October 30, 2024).

GitHub. AI4Bharat/IndicLID. https://github.com/AI4Bharat/IndicLID.

Ali Hürriyetoğlu, Hristo Tanev, Osman Mutlu, Surendrabikram Thapa, Fiona Anting Tan, and Erdem Yörük. 2023. Challenges and applications of automated extraction of socio-political events from text (CASE 2023): Workshop and shared task report. In *Proceedings of the 6th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text*, pages 167–175, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

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.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

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.

Deepak Kumar, Yousef Anees AbuHashem, and Zakir Durumeric. 2024. Watch your language: Investigating content moderation with large language models. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 18, pages 865–878.

Madhuri Kumbhar and Kalpana Thakre. 2024. Language identification and transliteration approaches for code-mixed text. *Journal of Engineering Science & Technology Review*, 17(1).

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.

Kritesh Rauniyar, Sweta Poudel, Shuvam Shiwakoti, Surendrabikram Thapa, Junaid Rashid, Jungeun Kim, Muhammad Imran, and Usman Naseem. 2023. Multiaspect 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)*.

Surendrabikram Thapa, Farhan Ahmad Jafri, Ali Hurriyetoğlu, Francielle Vargas, Roy Ka-Wei Lee, and Usman Naseem. 2023a. Multimodal hate speech event detection: Shared task 4, case 2023. In *Proceedings of the Workshop on Challenges and Applications of Social Media Analysis (CASE 2023)*. Association for Computational Linguistics.

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. 2023b. Nehate: Large-scale annotated data shedding light on hate speech in nepali local election discourse. In *ECAI 2023*, pages 2346–2353. IOS Press.

Bryan Wilie, Karissa Vincentio, Genta Indra Winata, Samuel Cahyawijaya, Xiaohong Li, Zhi Yuan Lim, Sidik Soleman, Rahmad Mahendra, Pascale Fung, Syafri Bahar, et al. 2020. Indonlu: Benchmark and resources for evaluating indonesian natural language understanding. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 843–857.

L. Zhang, S. Wang, and B. Liu. 2018. Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1253.