SKPD Emergency@NLU of Devanagari Script Languages 2025: Devanagari Script Classification using CBOW Embeddings with Attention-Enhanced BiLSTM

Shubham Shakya[1, 2], Saral Sainju[1, 3], Subham Krishna Shrestha[1, 4], Prekshya Dawadi[1, 5], Shreya Khatiwada[1, 6]

[1]Department of Computer Science and Engineering, Kathmandu University

Emails: {[2]ss46041720, [3]ss42041720, [4]ss50041720,[5]pd12041720, [6]sk29041720}@student.ku.edu.np

All authors contributed equally to this work.

Correspondence: shubham.shakyaa@gmail.com

Abstract

Devanagari script, encompassing languages such as Nepali, Marathi, Sanskrit, Bhojpuri and Hindi, involves challenges for identification due to its overlapping character sets and lexical characteristics. To address this, we propose a method that utilizes Continuous Bag of Words (CBOW) embeddings integrated with attention-enhanced Bidirectional Long Short-Term Memory (BiLSTM) network. Our methodology involves meticulous data preprocessing and generation of word embeddings to better the model's ability. The proposed method achieves an overall accuracy of 99%, significantly outperforming character level identification approaches. The results reveal high precision across most language pairs, though minor classification confusions persist between closely related languages. Our findings demonstrate the robustness of the CBOW-BiLSTM model for Devanagari script classification and highlights the importance of accurate language identification in preserving linguistic diversity in multilingual environments.

Keywords: Language Identification, Devanagari Script, Natural Language Processing, Neural Networks

1 Introduction

Devanagari Script, part of the Brahmic family of scripts, is widely used in regions such as India, Nepal, Tibet, and Southeast Asia (Mhaiskar, 2014). Often referred to simply as 'Nagari', it has historical significance, with some attributing the name to the writing system of 'city people', while others believe it originates from the Nagar Brahmans of Gujarat (Lambert, 1953). Today, Devanagari serves as the standardized writing system for several major South Asian languages, including Hindi, Nepali, Sanskrit, Bhojpuri, and Marathi. The volume of text data produced in these languages is substantial and continues to grow with the expansion of digital content.

In multilingual contexts, accurate language identification is a critical preliminary step for many natural language processing (NLP) systems. A system trained to classify Nepali text, for example, would struggle to handle Marathi documents effectively, and a Hindi-to-Bhojpuri translation system would likely fail if it were provided with Sanskrit data. This makes precise language identification essential for ensuring the proper functioning of NLP applications. However, identifying languages within the Devanagari script poses unique challenges. These languages share a large character set and exhibit many similarities, while having significant variation in writing styles and grammar (Kopparapu and Vijayalaxmi, 2014). This complexity creates obstacles to developing reliable language identification systems, particularly when distinguishing between languages like Bhojpuri and Hindi, which share extensive lexical overlap.

To address these challenges, this study, which is a part of the first task of the challenges in processing south asian languages (CHIPSAL) workshop at COLING'24, focuses on the identification of five languages—Nepali, Marathi, Sanskrit, Bhojpuri, and Hindi—within the Devanagari script. The goal is to develop a model capable of accurately distinguishing between these languages despite their close linguistic relationships. By implementing a bidirectional Long Short-Term Memory (BiLSTM) network (Graves et al., 2014), we aim to capture the contextual information crucial for distinguishing between languages in the same script. BiLSTM networks have demonstrated success in various NLP tasks, and their ability to process text in both forward and backward directions makes them particularly suited to tasks involving complex language structures.

As digital text in Nepali, Marathi, Sanskrit, Bhojpuri, and Hindi grows, accurate language identification becomes essential. Our study addresses this need by creating a robust framework for identify-

Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025), pages 339–343
January 19, 2025.

ing languages in the Devanagari script, highlighting the importance of preserving the linguistic diversity and cultural significance each language represents.

2 Related Works

This section reviews the existing study that addresses the challenges and methods used in language identification, mainly those that utilized the Devanagari scripts.

In the earlier period, the problem of language identification was approached by methods that utilized n-gram models. These models are foundational but have limitations when it comes to distinguishing between languages that have similar vocabularies.(Cavnar and Trenkle, 1994) showed the ability of n-gram models in language identification but acknowledged their shortcomings in highly overlapping languages. This limitation is particularly pronounced in Devanagari Scripts as different languages share extensive lexical similarities.

There has been a shift towards machine learning and deep learning techniques in order to improve the language identification accuracy. For example, (Joshi et al., 2020) analyzed the characters and word embeddings for Devanagari text classification, demonstrating the method including ResNet's success in recognizing the linguistic intricacies which performed better than the convolutional neural network which is the current state of the art.

Bidirectional Long Short-Term Memory networks have emerged as a promising avenue for solving the problem of Language Identification. (Bedyakin et al., 2021) work on offensive language identification in low-resource language using BiLSTM networks illustrates the model's effectiveness in handling unique linguistic characteristics. Our approach also aims to use BiLSTM networks for language identification between languages that share lexical similarities. Overall, the current research on language classification among the languages that use the Devanagari script is focused on developing robust and accurate techniques for script recognition, segmentation, and language identification. The diversity of Devanagari-based languages and the complexity of the script itself continue to drive research in this area. However, the specific task of classifying languages such as Hindi, Nepali, Marathi, Sanskrit, and Bhojpuri within the Devanagari script remains an unexplored area that requires further investigation.

3 Methodology

This section describes our approach to Devanagari script classification using CBOW embeddings with an attention-enhanced BiLSTM model. Our methodology comprises of four main components: data preprocessing, word embedding generation, neural network architecture and model training.

3.1 Data Preprocessing

In this study, we utilized publicly available datasets, includeing (Jafri et al., 2024), (Jafri et al., 2023), (Thapa et al., 2023), (Rauniyar et al., 2023), (Ojha, 2019), (Kulkarni et al., 2021), (Aralikatte et al., 2021) to ensure transparency and reproducibility. The preprocessing pipeline consists of several steps to clean and standardize the Devanagari text data.

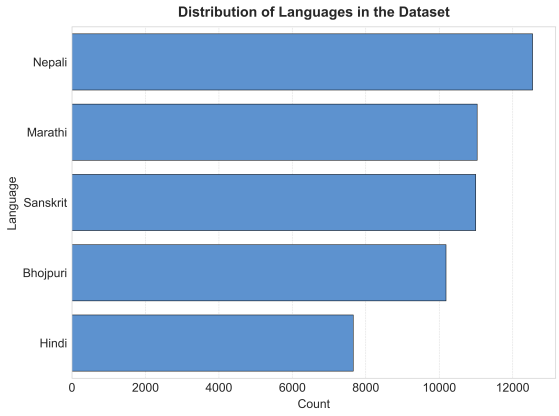

Figure 1: Distribution of Language in the Dataset

3.1.1 Text Cleaning

The dataset used for this research consists of 52,422 Devanagari-script text samples labeled by class as shown in Figure 1. First, the dataset was preprocessed to ensure data quality. Any missing text entries were removed. Non-Devanagari characters, numerals, and punctuation marks were eliminated using regular expressions. Each text entry was then stripped of whitespace resulting in a clean corpus of Devanagari text data.

3.1.2 Data Splitting

The preprocessed dataset was split into training and testing sets using an 80:20 ratio. For the final model training after hyper-parameter tuning, the entire dataset was used for model training. Evaluation was performed on a separate hold-out dataset.

Layer	Details
Input Layer	FastText Embeddings (Dimension: 100)
Bi-LSTM	LSTM (Input: 100, Hidden Units: 256, Layers: 3, Bidirectional: True)
Attention	Self-Attention (3 Linear Transformations: Wa, Ua, Va, Hidden Size: 512)
Output Layer	Fully Connected (Input: 512, Output: 5)

Table 1: Component configuration parameters for the model architecture.

3.2 Word Embedding Generation

In this research, two distinct methods for generating word embeddings were explored: Continuous Bag of Words (CBOW) model and a character-level encoding approach. Each method offers unique advantages and challenges in capturing the semantic richness of the Devanagari script.

3.2.1 Continious Bag of Words (CBOW)

The Continuous Bag of Words (CBOW) model is a predictive model used in natural language processing that captures contextual information by predicting a target word based on its surrounding words (Xia, 2023). In our implementation, we utilized FastText to train the CBOW model on a corpus of Devanagari text. The embedding dimension was set to 50 and embedding length of sentences was limited to 100 by either padding or truncating. This model generates dense vector representations of words which is effective in tasks like text classification and sentiment analysis (Xiong et al., 2019).

3.2.2 Character Level Encoding

Another approach explored in this research was character-level encoding. Each character of the text was converted into its Unicode code point, allowing for a straightforward numerical representation.We used this method to generate tensors of these code points, with a maximum sequence length of 100 characters. While this method provided a simplistic representation of the text, it did not capture the semantic relationships between characters as effectively as the CBOW embeddings which resulted in lower accuracy as shown in Table 4

3.3 Neural Network Architecture and Training

The Devanagari text classification model combines FastText embeddings with a Bidirectional Long Short-Term Memory (BiLSTM) network and an attention mechanism to capture features in each input sequence. The architecture is shown in table 1 First, each word in the text is transformed into a dense vector using FastText embeddings, which carry rich linguistic information. These vectors are processed by a bidirectional LSTM layer that captures context from both directions, essential for Devanagari script, where meaning is influenced by surrounding words. An attention layer then assigns weights to each word's hidden state to focus on the most relevant parts of the sequence, which allows the model to focus on specific parts of the input sequence (Zhang and Chu, 2023). This context vector is finally passed through a fully connected layer to output class probabilities. The model is trained using Cross-Entropy Loss and the Adam optimizer as shown in table 2, refining its weights across multiple epochs to improve accuracy in classifying Devanagari text.

Parameter	Value
Batch Size	32
Learning Rate	0.001
Epochs	10
Optimizer	Adam
Loss Function	Cross-Entropy

Table 2: Training parameters for the model.

4 Results

4.1 Model Performance

Our attention-enhanced BiLSTM model with CBOW embeddings demonstrated strong performance in Devanagari script classification. Figure 3 presents the detailed classification metrics for our proposed model.

4.2 Embedding Strategy Comparison

The results shown in table 4 demonstrate that both embeddings achieved comparable performance, significantly outperforming the character encoding approach. The superior performance of word embeddings can be attributed to their ability to capture semantic relationships and contextual information

Class	Precision	Recall	F1-Score	Support
Nepali	0.99	1.00	0.99	2688
Marathi	0.99	0.96	0.98	2365
Sanskrit	1.00	1.00	1.00	2356
Bhojpuri	0.96	0.99	0.97	2183
Hindi	0.94	0.93	0.93	1642
Accuracy	0.99			
Macro Avg	0.98	0.98	0.98	11234
Weighted Avg	0.98	0.98	0.98	11234

Table 3: Classification report showing precision, recall, f1-score, and support for each class.

Embedding Method	Accuracy
CBOW	99%
Skip-gram	97%
Character Encoding	72%

Table 4: Accuracy of different embedding methods.

in the Devanagari script, while character-level encoding fails to capture these higher-level linguistic patterns.

4.3 Error Analysis

We conducted a detailed error analysis to understand the model's classification behavior across different languages. Figure 2 presents the confusion matrix of our model's predictions.

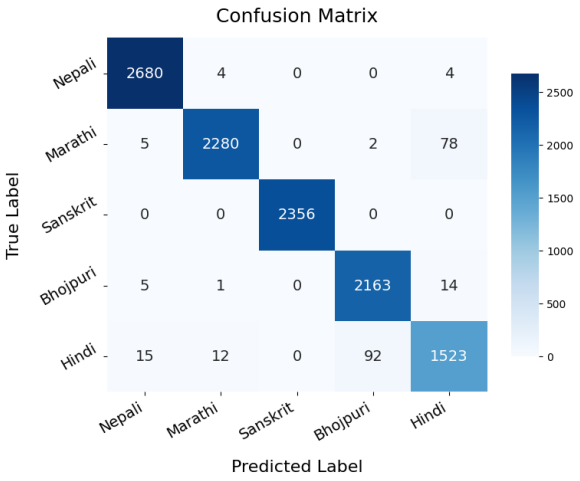

Figure 2: Confusion Matrix

Analysis of the matrix reveals several key patterns. Sanskrit achieved perfect classification with zero misclassifications, while Nepali showed robust performance with only 8 misclassifications. The most significant confusions occurred between Hindi-Bhojpuri (92 cases) and Marathi-Hindi (78 cases), likely due to their linguistic similarities. These patterns suggest that while the model excels at distinguishing languages with distinct characteristics, it faces some challenges with closely related language pairs.

4.4 Limitation and Future Enhancements

While the CBOW-BiLSTM model performs well, it faces challenges distinguishing closely related languages like Hindi and Bhojpuri due to their linguistic similarities. The character-level encoding method also underperforms compared to CBOW embeddings, as it lacks semantic depth. Future improvements could involve using transformer-based models to better handle these nuances and expanding the dataset to include more language variations and multi-modal data, such as handwritten text, to enhance accuracy and generalization.

5 Conclusion

This paper presents a practical and effective approach for identifying languages in the Devanagari script, a crucial task in today's multilingual world. By using CBOW embeddings combined with an attention-enhanced BiLSTM model, we show that our method can accurately distinguish between Nepali, Marathi, Sanskrit, Bhojpuri, and Hindi, providing a boost in precision over traditional techniques. Our findings highlights the value of a well-rounded preprocessing process and the role of attention mechanisms in improving performance for tasks involving Devanagari text. Overall, we believe our approach offers a flexible framework that can inspire further research, particularly in tackling the complex challenges of multilingual and mixed-language text in Devanagari.

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.

R. Bedyakin, M. Htsts, and N. Mikhaylovskiy. 2021. Low-resource spoken language identification using self-attentive pooling and deep 1d time-channel separable convolutions. *Proceedings of the 2021*, pages 1012–1020.

W. B. Cavnar and J. M. Trenkle. 1994. N-gram-based text categorization. In *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175.

A. Graves, N. Jaitly, and A. Mohamed. 2014. Hybrid speech recognition with deep bidirectional lstm. In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 273–278.

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.

A. Joshi, P. Solanki, and M. Joshi. 2020. Language identification of devanagari text using embedding-based approaches. In P. K. Pattnaik and S. Rautaray, editors, *Progress in Computing, Analytics and Networking*, pages 167–176. Springer, Singapore.

S. K. Kopparapu and K. Vijayalaxmi. 2014. *Challenges in Optical Character Recognition of Devanagari and Related Indic Scripts*. Springer, India.

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.

C. M. Lambert. 1953. A new study of the devanagari script: Its development and historical context. *Journal of the Royal Asiatic Society of Great Britain & Ireland*, 85(1):45–58.

M. Mhaiskar. 2014. A comprehensive study of brahmic scripts: Their origin, spread, and development. *Journal of South Asian Studies*, 30(2):123–145.

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

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)*.

N. Taghizadeh and H. Faili. 2020. Cross-lingual adaptation using universal dependencies. *arXiv*.

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.

H. Xia. 2023. Continuous-bag-of-words and skip-gram for word vector training and text classification. *Journal of Physics: Conference Series*, 2634:012052.

Zeyu Xiong, Qiangqiang Shen, Yueshan Xiong, YijieWang, and Weizi Li. 2019. New generation model of word vector representation based on cbow or skip-gram. *Computers, Materials amp; Continua*, 60(1):259–273.

H. Zhang and P. Chu. 2023. Prediction study based on tcn-bilstm-sa time series model. *Atlantis Highlights in Intelligent Systems*, pages 192–197.