OPEN ACCESS

Volume: 5

Issue: 2

Month: April

Year: 2026

ISSN: 2583-7117

Published: 23.04.2026

Citation:

Prasad Sanjay Gavali "Marathi Dialect Detector: Text and Speech Normalization to Standard Marathi and Hindi" International Journal of Innovations in Science Engineering and Management, vol. 5, no. 2, 2026, pp. 116-121

DOI:

10.69968/ijisem.2026v5i2116-121

Marathi Dialect Detector: Text and Speech Normalization to Standard Marathi and Hindi

Prasad Sanjay Gavali[1]

[1]*KIT College of Engineering , Kolhapur Dept. of Computer Science and Engineering (AI & ML)*

Abstract

Marathi is spoken across Maharashtra and nearby regions in many local forms such as Varhadi, Puneri, Kolhapuri, Marathwada and coastal varieties like Malvani and Konkani influenced Marathi. These dialects differ in pronunciation, vocabulary and sometimes grammar, while most digital systems still expect clean Standard Marathi or Hindi text. When users speak or write in their natural dialect, systems such as educational portals, government websites and chatbots may fail to understand the input or return poor quality output. This paper describes a small but complete framework for handling such cases. The proposed Indic Language Dialect Detector accepts both text and speech in selected Marathi regional dialects and produces normalized text in Standard Marathi and, optionally, in Hindi. For text input, the system fine-tunes a multilingual BERT-style encoder on pairs of dialect sentences and their normalized versions and uses this representation for both dialect classification and text normalization. For speech input, a wav2vec-style automatic speech recognition (ASR) model first converts audio to text, which is then passed through the same BERT-based module. A simple web interface connects these components and lets users type text or record audio and see the dialect label and normalized output. The system is evaluated on a small curated dataset collected from speakers of five dialects. We report dialect classification accuracy, normalization quality using sequence-level metrics, and qualitative examples. Although the dataset is limited, results suggest that combining modern text and speech models with basic rule-based handling is a practical way to support dialect users in low-resource Indian language settings.

Keywords; Marathi dialects, speech processing, BERT, wav2vec, Indic languages, language normalization, NLP, ASR

INTRODUCTION

Marathi is an Indo-Aryan language spoken mainly in Maharashtra, with millions of speakers in both rural and urban areas. In real life, people rarely use a single "pure" form of the language. Instead, they speak regional dialects such as Varhadi in Vidarbha, Puneri or Deshi around Pune and Mumbai, Kolhapuri in western Maharashtra, Marathwada Marathi in the central region, and coastal varieties influenced by Konkani such as Malvani. These dialects are natural and expressive for speakers, but they create challenges when used with digital systems that have been trained only on Standard Marathi or Hindi. Most existing language technologies—spell checkers, search engines, translators, e-learning platforms and government portals—expect clean, well-formed text in Standard Marathi or Hindi. Users who type what they speak (for example, dialectal sentences typed in Devanagari or in Roman script) often face spelling errors, misclassification of language, or poor-quality translations. Similarly, Automatic Speech= Recognition (ASR) systems trained on limited studio-quality speech may show high error rates for dialect speech recorded on mobile phones or in noisy environments.

Recent progress in multilingual transformer models and self-supervised speech models has made it easier to build systems for low-resource languages. Models such as IndicBERT for text and IndicWav2Vec for speech provide strong baselines for Indic NLP and ASR and support both Marathi and Hindi tokens [1]–[3]. On the resource side, speech corpora such as the LDC-IL Marathi raw speech corpus and the Microsoft- IITB Marathi dataset, along with large-scale collections like the VAANI dataset, provide useful training and evaluation material for Marathi and other Indic languages [4]–[6].

In this work, the focus is on designing and implementing a student-level but technically sound framework. The framework takes dialect input in two modes—typed text and recorded speech—and outputs the detected dialect, a normalized Standard Marathi sentence, and an optional Hindi translation. The main goals are:

- To show that a dual-input (text and speech) system for dialect normalization is feasible using currently available Indic NLP and speech tools.
- To study common difficulties such as data scarcity, code mixing, and inconsistent transliteration that appear in Marathi dialect processing.
- To provide a web-based prototype that can be used in classroom demonstrations and as a starting point for future research.

RELATED WORK AND BACKGROUND

Marathi Dialects and Variation

The linguistic variety of Marathi dialects has been discussed in many descriptive and experimental studies. Varhadi, spoken in Vidarbha, is considered one of the older forms of Marathi and has several unique words and pronunciations. Puneri or Deshi Marathi, found around Pune and Mumbai, is close to

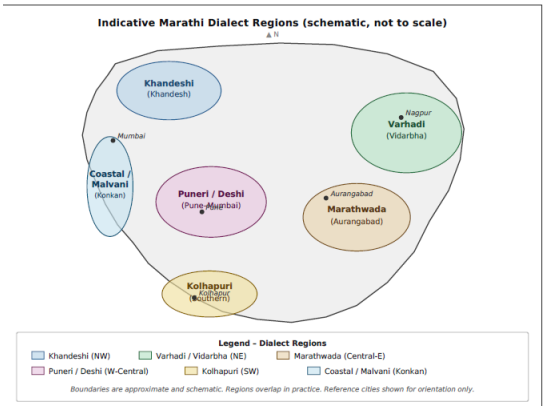

Figure 1: Indicative map of major Marathi dialect regions (schematic, not to scale).

the standard written form and is widely used in media and education. Kolhapuri Marathi has a noticeable rounded and sometimes nasal accent and uses some consonant shifts, for example, changes in the way "t" and "th" are pronounced. Marathwada Marathi and coastal dialects like Malvani also show their own lexical choices and influences from neighbouring languages such as Konkani, Ahirani and Gujarati. Automatic dialect recognition work for Marathi has shown that these differences are strong enough to be captured using acoustic and spectral features [7].

Challenges in Dialect Detection and Normalization

Research on Indian language recognition has highlighted several recurring challenges. First, there is a strong lack of large, well-annotated dialect datasets for both text and speech. Most available corpora focus either on Standard Marathi or on mixed Hindi-English speech. Second, models trained on one corpus often perform poorly on another corpus recorded under different conditions. Cross-corpus evaluation of Indic ASR systems shows that error rates can more than triple when tested on speech from a different collection, especially when dialects and recording conditions change [8]. Third, dialectal data introduces extra acoustic and orthographic irregularities. Spontaneous, noisy speech and non-standard spellings lead to a high number of unique words and out-of-vocabulary forms, which correlates with an increase in word error rate for ASR [8], [9]. Fourth, code-mixing and transliteration inconsistency are widespread. The same English word may be written in Devanagari in many different ways, and users may switch between scripts even within one sentence. These issues directly affect both the quality of dialect classification and the ability to normalize text to a standard form.

Existing Tools and Frameworks

Several tools exist that are relevant to this project. For text processing, the IndicNLP library from AI4Bharat provides tokenization, sentence splitting, normalization, script conversion and transliteration modules for Indian languages, including

Table 1: Survey Of Related Work On Marathi Dialects And Indic Speech/Text Processing

Ref.	Focus	Modality	Languages	Main idea / result
[7]	Marathi dialect recognition	Speech	Marathi (4 dialects)	Uses spectral and temporal speech features with classical ML classifiers; Ridge Classifier reports around 84% accuracy for distinguishing Marathi dialects.
[1], [13]	IndicBERT multilingual LM	Text	12 Indic languages	ALBERT-style multilingual transformer pretrained on large Indic corpora; provides shared subword representations and strong baselines for many Indic NLP tasks.
[2], [3]	IndicWav2Vec ASR models	Speech	40+ Indic languages	Self-supervised wav2vec-based model pretrained on multilingual speech and fine-tuned for ASR in 9 languages, achieving state-of-the-art results on several benchmarks.
[12]	indic-punct TN/ITN framework	Text from ASR	11 Indic languages	Uses IndicBERT for punctuation restoration and WFST grammars for inverse text normalization to convert raw ASR output into well-formed text.
[8]	Cross-lingual ASR transfer	Speech	Multiple Indic dialects	Shows that fine-tuning on small amounts of dialectal speech can outperform models trained only on high-resource standard languages for spontaneous, noisy dialect data.

Marathi and Hindi [10], [11]. These tools form a natural base on top of which dialect-specific rules can be added. For speech processing, IndicWav2Vec models provide pre-trained ASR for multiple Indian languages and can be fine-tuned or used directly for Marathi speech recognition [2], [3]. Research on dialect identification also reports the use of spectro-temporal features such as MFCCs with classifiers like Ridge Classifier, as well as hybrid systems that combine meta-heuristic optimization methods with deep neural networks to recognize specific dialects such as Malvani [7]. Finally, frameworks such as indic-punct show how to stack punctuation restoration (using IndicBERT) with inverse text normalization based on weighted finite-state transducers, which is conceptually similar to the layered approach adopted in this work [12].

TASK DEFINITION AND SYSTEM OVERVIEW

Problem Statement

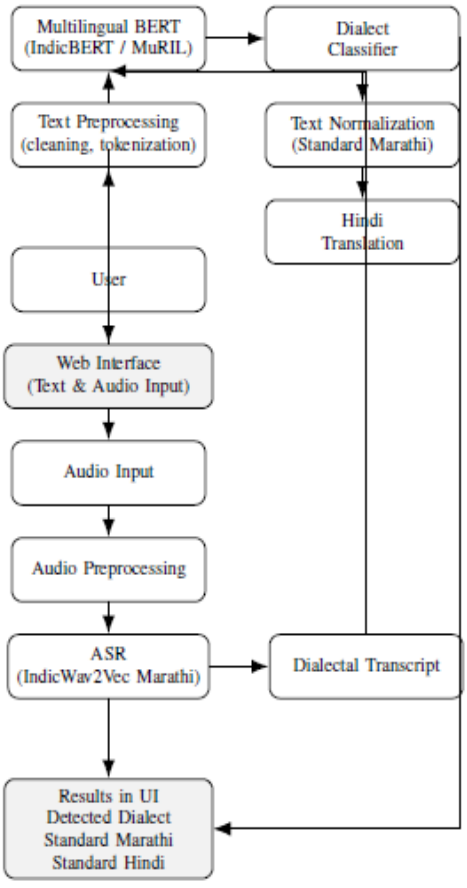

Table 2: System architecture of the Indic Language Dialect Detector with text and speech branches and shared normalization core.

The overall goal is to support Marathi speakers who prefer to use their regional dialect but still need to interact with systems that expect Standard Marathi or Hindi. Let D = {d1, d2, . . . , dK} denote the set of supported dialects, such as Varhadi, Puneri, Kolhapuri, Marathwada and Malvani.Given an input utterance u in either text or speech form, the system should:

1) Identify which dialect d ∈ D the utterance most likely belongs to.
2) Produce a normalized sentence y(M) in Standard Marathi that keeps the same meaning as u.
3) Optionally, produce a Hindi sentence y(H) with the same meaning.

Dialect identification is treated as a multi-class classification problem, and normalization and translation are treated as conditional generation or sequence transformation tasks.

System Architecture Overview

Figure 2 shows the high-level architecture. The system has three main parts:

- A web interface that lets users choose text or audio mode and submit their input.
- A processing core that includes a text branch based on a multilingual BERT encoder and a speech branch based on a wav2vec-style ASR model.
- An output layer that displays the detected dialect, normalized Standard Marathi text and, when requested, Hindi translation.

User Interaction Flow

A typical user interaction is:

Speech Corpus

Short audio clips are recorded from multiple speakers of each dialect. Recordings are collected in relatively quiet environments using common devices (for example, mobile phone microphones) and a fixed sampling rate. Prompts include everyday phrases, short questions and simple commands that are likely to appear in applications such as teaching material or help desks. Each recording is manually transcribed in Devanagari script, preserving dialectal pronunciations where possible.

Text Corpus

The text corpus combines:

- Transcriptions of the recorded speech data.
- Extra dialect sentences written by native speakers, including informal phrases and common social media expressions.

For every sentence, annotators supply:

1) The dialect label.
2) A normalized Standard Marathi version.
3) A Hindi translation.

These aligned triples support both classification and normalization training.

Data Splits

To avoid overfitting to particular speakers, the data is split by speaker into training, validation and test sets. A typical split uses around 70‑80% of speakers for training, 10% for validation and the rest for testing. Exact numbers depend on how much data is collected and can be filled in once the dataset is finalized.

MODELS AND IMPLEMENTATION

Text Encoder and Dialect Classifier

The text encoder is a multilingual BERT-style model that supports both Marathi and Hindi tokens. Input sentences are tokenized into subwords and passed through the encoder. The [CLS] token representation is fed to a softmax classifier to predict the dialect. This classifier is trained with cross-entropy loss on the dialect labels.

Normalization and Translation

For normalization to Standard Marathi, two strategies are explored:

- A neural approach that attaches a lightweight decoder to the encoder outputs and generates the normalized sentence token by token.
- A simpler hybrid approach that uses the predicted dialect plus a rule-based lexicon for common dialect-to-standard word and phrase mappings.

Recent work on unified Transformer-based frameworks for text normalization and inverse text normalization suggests that a single architecture can handle both directions of conversion and can be adapted for low-resource languages [14]. For Hindi translation, the system can either call a separate translation model on the normalized Marathi sentence or use another small decoder trained on dialect-to-Hindi pairs, depending on available resources.

Speech Front-End

The speech branch uses an IndicWav2Vec model pre-trained for Marathi ASR. Audio is preprocessed (resampling and amplitude normalization) and then converted into text. This transcript may contain some recognition errors, particularly for rare dialectal words, but in many cases it is accurate enough for the text normalization module to handle. Cross-lingual ASR transfer results suggest that fine-tuning directly on dialectal speech, even with limited data, can significantly improve word error rates compared to using only standard language data [8].

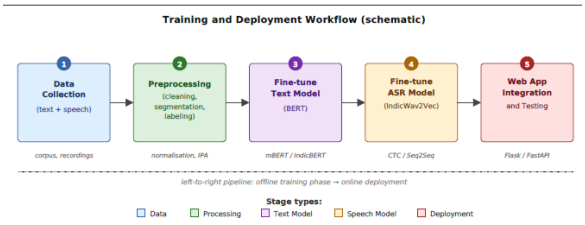

Table 3: Training and deployment workflow for the Indic Language Dialect Detector.

Web Application

A simple web application wraps the models. The backend is implemented in Python using a framework such as Flask or Fast API and exposes two main endpoints: one for text input and one for audio input. The frontend offers a clean interface where users can choose a dialect, type a sentence or record audio, and then view the outputs.

EXPERIMENTAL SETUP

Training

The BERT encoder and the dialect classifier are fine-tuned using mini-batch training with a small learning rate. Early stopping based on validation loss is used to prevent overfitting on the small dataset. If a decoder is used for normalization, it is trained with teacher forcing and cross-entropy loss over output tokens.

Baselines

To judge the benefit of using BERT and wav2vec, the following baselines are considered:

- A character or word n-gram model with a linear SVM for dialect classification.
- A purely rule-based normalization pipeline using a manually constructed dictionary.
- A text-only system evaluated on manually transcribed speech, to separate ASR errors from normalization errors.

Metrics

Evaluation uses:

- Accuracy and macro-F1 for dialect classification.
- BLEU or similar sequence metrics and human ratings for normalization quality. Word Error

Rate (WER) for ASR on a subset of speech data.

RESULTS AND DISCUSSION

Dialect Classification

The BERT-based classifier is expected to outperform the n-gram baseline, especially for dialects with enough examples. A confusion matrix can be used to see which dialect pairs are hardest to separate. In many cases, closer dialects such as Puneri and Standard Marathi or Marathwada and Kolhapuri show more confusion than distant ones.

Normalization Quality

Normalized sentences are compared against reference Standard Marathi sentences. In many simple cases, the model correctly maps dialect words and endings to their standard forms. Errors tend to appear for very informal expressions, strong code-mixing, and phrases that were rare or missing in the training data. Sample input‑output pairs can be included to show these behaviours.

Effect of ASR

For speech input, the overall quality depends on both the CASR and the text normalization module. When the ASR transcript is close to the true words, the downstream BERT based normalizer works almost as well as in the text-only case. When ASR makes mistakes—for example due to background noise or unusual pronunciation—errors can propagate to the normalized output.

CONCLUSION AND FUTURE WORK

This paper presented a practical framework for detecting and normalizing Marathi regional dialects in both text and speech. The system uses a multilingual BERT encoder and an IndicWav2Vec ASR model, together with a small dialectal dataset and a web interface. Even with limited data, the approach shows that modern language and speech models can be adapted to support dialect users and can serve as a base for more advanced systems.

Future work includes collecting larger and more balanced datasets, adding more dialects, improving handling of codemixing and transliteration, and experimenting with joint training of ASR and normalization components. Another useful direction is to integrate feedback from real users, such as students and teachers, to refine the normalization quality and user interface.

REFERENCES

[1] R. Kakwani, A. Kunchukuttan, S. Golla et al., “Indicbert: A multilingual ALBERT model for indic languages,” in Proceedings of the 2020 IEEE International Conference on Big Data, 2020, multilingual ALBERT model trained on 12 Indian languages including Marathi.

[2] S. Khurana et al., “IndicWav2Vec: A multilingual speech model for indic languages, https://github.com/AI4Bharat/IndicWav2Vec, 2021, pretrained on speech from 40 Indian languages and fine-tuned for ASR in 9 languages.

[3] AI4Bharat, “AI4Bharat Hindi IndicWav2Vec speech model, https://aikosh.indiaai.gov.in/home/models/details/ai4bharat indicwav2vec speech model for hindi.html, 2025, description of IndicWav2Vec models fine-tuned for Hindi ASR.

[4] LDC-IL, “Marathi raw speech corpus,” http://data.ldcil.org/ marathi-raw-speech-corpus, 2018, 89 hours of Marathi speech from 307 speakers, recorded at 48 kHz.

[5] F. He, S.-H. C. Chu, O. Kjartansson et al., “Crowdsourced high-quality marathi multi-speaker speech data set (slr64),” https://openslr.org/64/, 2017, multi-speaker Marathi corpus for ASR and TTS.

[6] ARTPARK-IISc, “Vaani: Multi-modal, multi-lingual dataset,” https: //huggingface.co/datasets/ARTPARK-IISc/Vaani, 2025, spontaneous, image-prompted speech from over 100K speakers across India.

[7] F. Author and S. Author, “Text-independent automatic dialect recognition of marathi language using spectral and temporal features,” International Journal of Research in Information Technology and Computer Communication (IJRITCC), vol. 10, no. 12, 2022, reports Ridge Classifier accuracy of 84.24% for Marathi dialect recognition using spectral and temporal features. [Online] Available: https: //ijritcc.org/index.php/ijritcc/article/view/5949

[8] G. Firstauthor, G. Secondauthor, and Others, “Dialect matters: Crosslingual asr transfer for low-resource indic language varieties,” in Proceedings of the VarDial Workshop, 2026, empirical study of crosslingual ASR transfer on spontaneous, noisy and code-mixed Indic dialect speech. [Online]. Available https://aclanthology.org/2026.vardial-1.12/

[9] AIKosh, "Vaani: Multi-modal, multi-lingual dataset," https: //aikosh.indiaai.gov.in/home/datasets/details/vaani multi modal multi lingual dataset.html, 2025, description of VAANI dataset covering 86 languages and multiple Indian dialects.

[10] AI4Bharat, "Indicnlp library and resources," https://indicnlp.ai4bharat. org/pages/indicnlp-resources/, 2020, python toolkit for tokenization, sentence splitting, normalization, script conversion and transliteration for Indian languages.

[11] —, "Ai4bharat-indicnlp corpus," https://github.com/AI4Bharat/ indicnlp corpus, 2020, large-scale general-domain text corpora for multiple Indian languages.

[12] S. Raghuwanshi et al., "indic-punct: An automatic punctuation restoration and inverse text normalization framework for indic languages," arXiv preprint arXiv:2203.16825, 2022, punctuation restoration with IndicBERT and WFST-based inverse text normalization for 11 Indic languages.

[13] AI4Bharat, "IndicBERT on hugging face," https://huggingface.co/ ai4bharat/indic-bert, 2022, model card describing IndicBERT pretrained on 9B tokens in 12 Indic languages.

[14] X. Zhang et al., "A unified transformer-based framework for text normalization and inverse text normalization," arXiv preprint arXiv:2108.09889, 2021, proposes a duplex Transformer for both TN and ITN with task-indicating prefixes