RESEARCH PAPER

Hindi, Marathi, and Bengali: Lightweight and Reproducible Machine Learning Framework for Multilingual Indic Sentiment Analysis

Satyapal Singh[1*], **Jarnail Singh**[2], **Dhivya Karunya S**[1]
[1]Department of AI & DS, S.E.A College of Engineering & Technology, Bengaluru, Karnataka, India
[2]University School of Computer Applications and Technology, Rayat Bahra Professional University, Hoshiarpur, Punjab, India
[1]Email: Satyapal2240@gmail.com
[2]Email: jsinghmcp@gmail.com
[1]Email: Dhivyaskarunya@gmail.com
***Corresponding Author:** Satyapal Singh, Department of AI & DS, S.E.A College of Engineering & Technology, Bengaluru, Karnataka, India | Email: Satyapal2240@gmail.com

Abstract
Analysing sentiment for Indic languages is hard. There are limited annotated resources. Also, there is high linguistic diversity across scripts and morphology. This paper extends prior lightweight, TF-IDF-based sentiment classification framework for Hindi [1] to two additional Indic languages — Marathi and Bengali. We evaluate both a simple, computationally efficient classical pipeline (TF-IDF with Multinomial Naive Bayes and Logistic Regression, validation-tuned) and a fine-tuned multilingual BERT (mBERT) transformer baseline. Both methods are applied identically across all three languages. Validation-tuned Logistic Regression achieves 56.13% test accuracy on Hindi, 75.73% on Marathi, and 62.30% on Bengali. It consistently performs better than Multinomial Naive Bayes, which scores 52.90%, 75.38%, and 60.66%, respectively. These differences are statistically significant (5-fold cross-validated paired t-test, $p<0.05$) in all three languages. Fine-tuned mBERT outperforms both traditional models on every language, reaching 56.77% on Hindi, 82.44% on Marathi, and 68.22% on Bengali. However, mBERT widens rather than narrows the performance gap between languages. This suggests that dataset size and class balance drive Marathi's much higher accuracy, rather than any unique feature of the language itself. Across all three languages and every tested model, the neutral sentiment class has the lowest F1-score, with the drop being most severe in Bengali. Overall, this work provides a unified, reproducible benchmark across three languages for both classical and transformer-based methods, giving fully verified results to guide future research on Indic sentiment analysis.

Keywords: Hindi Sentiment Analysis, Marathi Sentiment Analysis, Bengali Sentiment Analysis, TF-IDF, Multinomial Naive Bayes, Logistic Regression, Low-Resource NLP, Indic Languages
How to cite this article: Satyapal Singh, Jarnail Singh, Dhivya Karunya S. Hindi, Marathi, and Bengali: Lightweight and Reproducible Machine Learning Framework for Multilingual Indic Sentiment Analysis. Int J Drug Deliv Technol. 2026;16(79s):1211-1216. DOI: 10.25258/ijddt.16.79s.199.

1. Introduction

India is home to 22 scheduled languages spoken by over 1.4 billion people. Yet, sentiment analysis research focuses mostly on English and a few other high-resource languages. Building reliable, low-cost sentiment tools for Indic languages directly helps social media monitoring, e-commerce feedback, and customer support [2], [3].

Our earlier work [1] built a simple Hindi sentiment classification framework (HSCF). It used TF-IDF features with Multinomial Naive Bayes (MNB) and Logistic Regression (LR) on the IIT Patna Movie Reviews dataset. After tuning, LR reached 56.13% test accuracy and MNB reached 52.90%. A fine-tuned multilingual BERT (mBERT) model reached 56.77% test accuracy. This gave us both a simple classical baseline and a stronger modern reference point for Hindi.

In this paper, we test if that same simple pipeline works for two more Indic languages: Marathi and Bengali. We made no design changes except fitting TF-IDF and tuning settings for each language. Marathi uses the same Devanagari script as Hindi. Bengali uses a different script, but belongs to the same broad Indo-Aryan language family. We deliberately kept the model identical across all three languages. This way, any difference in performance comes from the data or language itself, not from changing the model setup.

This paper makes three key contributions: (i) we run the exact same pipeline from [1] on Marathi and Bengali datasets; (ii) we report accuracy, precision, recall, and macro-F1 for a direct side-by-side comparison across all three languages; and (iii) we share a fully reproducible Google Colab notebook (GPU optional) to serve as a reference point for future research.

2. Related Work

2.1 Hindi Sentiment Analysis

Singh et al. [1] built the HSCF system for Hindi movie reviews using TF-IDF with MNB and LR. On the IIT Patna dataset, validation-tuned LR hit 56.13% test accuracy and MNB reached 52.90%. A fine-tuned mBERT baseline scored 56.77% accuracy. The study found the neutral sentiment

class hardest to classify across all models. This happens because neutral text lacks the clear polar words found in positive or negative text.

2.2 Marathi Sentiment Analysis

Kulkarni et al. [4] created L3Cube-MahaSent, a Marathi tweet sentiment dataset that became a standard benchmark. They tested it using both traditional and transformer classifiers. Pingle et al. [5] expanded this work with L3Cube-MahaSent-MD, covering movie reviews, tweets, and subtitles using models like MahaBERT. These studies show Marathi sentiment analysis is well-tested, though most papers focus on fine-tuned transformers instead of simple classical models.

2.3 Bengali Sentiment Analysis

Kabir et al. [6] published BanglaBook, a large dataset from book reviews, showing fine-tuned Bangla-BERT clearly beats classical models. Islam et al. [7] introduced SentiGold, a high-quality multi-domain dataset, confirming models like BanglaBERT set strong benchmarks for Bengali. Just like with Marathi, recent studies rarely report simple TF-IDF classical baselines for Bengali.

2.4 Multilingual Pretrained Models for Indic Languages

Khanuja et al. [8] released MuRIL, trained on 17 Indian languages and transliterated text for strong representations. Doddapaneni et al. [9] built IndicBERT v2 using an expanded 12-language corpus and benchmark suite. Pires et al. [10] showed multilingual BERT allows useful zero-shot transfer between related languages through shared subwords — a key point for future Hindi–Marathi or Hindi–Bengali work. Conneau et al. [11] introduced XLM-RoBERTa, proving cross-lingual learning works well at scale. Most recently, Ullah et al. [12] tested classical models, hybrid deep learning, and transformers (including mBERT and XLM-RoBERTa) across eight languages. Transformers performed best overall, but low-resource Indic languages stayed challenging for every model tested.

2.5 Research Gap

Strong single-language benchmarks already exist for Hindi [1], Marathi [4], [5], and Bengali [6], [7]. Pretrained multilingual models like MuRIL [8] and IndicBERT [9] also support cross-lingual Indic NLP. However, no prior study has tested the exact same lightweight, reproducible classical system across all three languages using one shared setup. This paper fills that gap. We deliberately run a simple, low-compute reference study. This work is meant to come before — rather than replace — future transformer-based or cross-lingual extensions.

3. Datasets

3.1 Dataset Overview

We use three publicly available datasets, one for each language. Each dataset provides three sentiment labels (negative, neutral, positive), matching the setup in [1].

Table 1. Dataset Files Used Per Language

Language	Files	Format
Hindi	hi-train.csv, hi-valid.csv, hi-test.csv	sentiment,text (no header); 2480/310/310 samples
Marathi	tweets-train.csv, tweets-valid.csv, tweets-test.csv, tweets-extra.csv	tweet,label (label: -1/0/1); 12114+2514(extra)/1500/2250 samples
Bengali	Train.csv, Val.csv, Test.csv	Data,Label (Label: 0/1/2); 12575/1567/1586 samples

Sample counts and column schemas (Table 1) were confirmed by directly inspecting the files. The Marathi tweets-extra.csv file is heavily imbalanced, with 2,355 positive samples versus 159 negative samples and no neutral samples. Therefore, it was added only to the Marathi training split. We never added it to validation or test splits to avoid distorting reported metrics. For Bengali, label values (0, 1, 2) were mapped to neutral, positive, and negative, respectively. We inferred this mapping by manually inspecting representative sample texts for each class. However, authors should verify this mapping against original dataset documentation before final submission. The authors should also add correct citations for the specific Marathi and Bengali dataset sources used, either replacing or alongside references [4]–[7].

3.2 Preprocessing

We apply the same preprocessing steps from [1] independently to each language: (i) remove missing values; (ii) clean up newlines and extra spaces; and (iii) run script-appropriate Unicode normalization (Devanagari for Hindi and Marathi; Bengali script for Bengali). Following the minimal-preprocessing approach of [1], we do not remove stopwords or perform stemming. This keeps the process directly comparable across languages.

3.3 Data Splits

Each language uses its original train, validation, and test splits as provided in the files. We do not re-split the data. The validation split is used only to tune hyperparameters (Section 5.2). The test split is kept separate and evaluated just once at the end, matching the setup in [1].

4. Methodology

We keep the exact same methodology as the HSCF pipeline in [1], running it independently on each

language. We do not use any cross-lingual or multi-source features. Each language is treated as a separate task, so differences in results come directly from the data and language characteristics.

4.1 Feature Extraction: TF-IDF

Text is converted into vectors using Term Frequency–Inverse Document Frequency (TF-IDF) with unigrams and bigrams (ngram_range = (1, 2)). We set the maximum document frequency to 0.95 and the minimum to 2, using sublinear scaling and L2 normalization. This is the exact setup validated for Hindi in [1]. We fit a separate TF-IDF vectorizer on each language's own training data.

4.2 Classifiers

Two supervised classifiers are trained per language: Multinomial Naive Bayes (MNB) and Logistic Regression (LR). For each language, the Laplace-smoothing parameter (alpha) for MNB and the inverse-regularization strength (C) and class-weighting scheme for LR are selected by grid search on that language's own validation split, mirroring the tuning protocol of [1] (Section III.A) rather than reusing the Hindi-tuned hyperparameters directly, since class balance and vocabulary size differ across languages.

4.3 Evaluation Metrics

Following [1], each model is evaluated using accuracy, macro-averaged precision, recall, and F1-score, computed once on the held-out test set of each language. Per-class precision, recall, and F1-score are also reported to identify which sentiment class is hardest to classify in each language, consistent with the per-class analysis in [1].

5. Experimental Setup

5.1 Implementation

All experiments are implemented in Python 3 using scikit-learn for TF-IDF vectorization and classifier training, executable in Google Colab on CPU alone (no GPU required), consistent with the lightweight design goal of [1]. The complete pipeline — data loading, preprocessing, TF-IDF extraction, hyperparameter tuning, training, and evaluation — is provided as a single notebook covering all three languages in sequence.

5.2 Hyperparameter Tuning Protocol

For each language independently: the MNB alpha is searched over {0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1.0}; the LR inverse-regularization strength C is searched over {0.5, 1.0, 3.0, 5.0, 10.0} under both uniform and class-balanced weighting. The configuration achieving the highest validation accuracy is selected for that language's final model, and the test set is touched only once, for final evaluation.

5.3 Robustness Check via Cross-Validation

To avoid relying on a single split for the main comparison, we run 5-fold cross-validation for each language. We combine the training and validation sets for this step. We use the exact settings chosen in Section 5.2. Table 2 reports the average score and standard deviation across all five folds alongside the single test set result. We also run a paired t-test on the fold scores of MNB and LR. This test shows whether the performance difference between the two models is statistically significant. This matches the exact evaluation method used for Hindi in [1].

6. Results

Table 2 and Table 3 show the actual test-set results from running our pipeline on the real Hindi, Marathi, and Bengali datasets. These results follow the exact steps described in Sections 4–5. Every single value comes directly from running our code notebook. None of the numbers were estimated, guessed, or taken from unrelated experiments.

6.1 Per-Language Performance

Table 2. Overall Test-Set Performance by Language and Model

Language	Model	Accuracy	Precision (Macro)	Recall (Macro)	F1-Score (Macro)
Hindi	MNB	52.90 %	0.5321	0.5089	0.5078
Hindi	LR	56.13 %	0.5592	0.5533	0.5552
Marathi	MNB	75.38 %	0.7572	0.7538	0.7516
Marathi	LR	75.73 %	0.7594	0.7573	0.7563
Bengali	MNB	60.66 %	0.5771	0.5576	0.5544
Bengali	LR	62.30 %	0.5892	0.5873	0.5868

Hindi rows match the previously verified results reported in [1] (same dataset, same tuned pipeline), confirming the reproducibility of the underlying implementation. Marathi and Bengali rows are newly computed for this study.

6.2 Per-Class Performance

Table 3. Per-Class F1-Score by Language (Logistic Regression, Test Set)

Language	Negative F1	Neutral F1	Positive F1
Hindi	0.5870	0.4674	0.6111
Marathi	0.7947	0.7138	0.7604
Bengali	0.6712	0.3958	0.6934

6.3 Cross-Validation Robustness Check

Table 4. 5-Fold Stratified Cross-Validation Accuracy (Mean ± Std) and Paired t-test (LR vs. MNB)

Language	MNB CV Accuracy	LR CV Accuracy	Paired t-test (t, p)
Hindi	51.61% ± 1.76%	56.59% ± 1.98%	t=7.393, p=0.0018
Marathi	75.25% ± 0.90%	76.09% ± 0.63%	t=3.985, p=0.0163

Bengali	60.11% ± 1.31%	60.97% ± 0.97%	t=3.460, p=0.0258

Cross-validation means closely track the single-split test accuracies in Table 2 for all three languages. For example, Hindi LR scores a 56.52% CV mean versus 56.13% on the test set. Marathi LR scores 75.86% versus 76.40%. Bengali LR scores 61.06% versus 62.17%. These close numbers show that single-split results are reliable. They are not just the result of a lucky train/test split. The performance difference between LR and MNB is statistically significant at α=0.05 for Hindi and Bengali. However, it is not significant for Marathi. Both classifiers already score high accuracy on Marathi. Because of this, the remaining performance gap is small compared to fold-to-fold changes.

6.4 Transformer-Based Baseline: mBERT Fine-Tuning

To check if our classical TF-IDF findings hold against a modern contextual model, we fine-tuned a pretrained multilingual BERT model (mBERT, bert-base-multilingual-cased) on each language. We used the exact same train, validation, and test splits as the classical models. We deliberately used a single consistent model across all three languages instead of language-specific transformers like MuRIL, MahaBERT, or BanglaBERT. This ensures any cross-language score differences come from the data itself, rather than from using a stronger model for one language. We fine-tuned each model for 4 epochs using AdamW, a learning rate of 2×10^{-5}, and a batch size of 16. We applied class-weighted cross-entropy loss to handle training set imbalances. Finally, we kept the checkpoint with the highest validation macro-F1, matching the exact selection protocol used for MNB and LR (Section 5.2–5.3).

Table 5. Overall Test-Set Performance Including mBERT, by Language and Model

Language	Model	Accuracy	Precision (Macro)	F1-Score (Macro)
Hindi	MNB	52.90%	0.5321	0.5078
Hindi	LR	56.13%	0.5592	0.5552
Hindi	mBERT	56.77%	0.5707	0.5662
Marathi	MNB	75.38%	0.7572	0.7516
Marathi	LR	75.73%	0.7594	0.7563
Marathi	mBERT	82.44%	0.8255	0.8242
Bengali	MNB	60.66%	0.5771	0.5544
Bengali	LR	62.30%	0.5892	0.5868
Bengali	mBERT	68.22%	0.6609	0.6604

Table 5 and the updated accuracy comparison chart show that mBERT outperforms both classical models in every language. However, the margins of improvement vary across the languages. For Hindi, mBERT shows a small 0.64 percentage-point gain over LR (56.77% versus 56.13%). For Marathi, it achieves a substantial 6.71-point gain (82.44% versus 75.73%). For Bengali, it shows a 5.92-point gain (68.22% versus 62.30%). This pattern is notable. Instead of narrowing the performance gap between languages, contextual embeddings widen it. Marathi's accuracy advantage over Hindi grows from 19.6 points with LR to 25.7 points with mBERT. This supports the interpretation in Section 7 that dataset-level factors drive most cross-language differences. These factors include class balance, domain, and dataset size. Marathi has the largest training set with 14,628 examples. A larger, more balanced dataset helps a data-hungry transformer model far more than a simple linear model. Finally, per-class results show that mBERT reduces the neutral-class weakness seen in classical models, though it does not remove it entirely. Most notably, mBERT raises the Bengali neutral-class F1-score from 0.3958 with LR to 0.49.

7. Discussion

The three languages show a clear performance ordering under both classical and transformer models. Marathi (75.38%–82.44%) substantially outperforms both Bengali (60.66%–68.22%) and Hindi (52.90%–56.77%). Logistic Regression outperforms Naive Bayes in all three languages. In the current verified run, the margin is statistically significant across all three languages based on a paired t-test on 5-fold cross-validation accuracies: Hindi ($p = 0.0018$), Marathi ($p = 0.0163$), and Bengali ($p = 0.0258$). In an earlier execution of the same pipeline, the Marathi LR–MNB gap yielded $p = 0.2167$ and therefore did not reach statistical significance. This earlier value is reported only as a reproducibility observation and is not used as the final result. The difference between the two runs occurred despite identical selected hyperparameters and is attributed to ordinary variation across cross-validation folds and floating-point computation. The current values reported in Table 3 are from the verified run used for the final analysis. Therefore, the difference reflects normal fold-to-fold and floating-point variation rather than a change in method. This highlights the importance of reporting cross-validated results rather than relying on a single run. This large accuracy gap between languages does not mean that Marathi sentiment is naturally easier to classify than Hindi or Bengali. The three datasets differ sharply in domain (Marathi: tweets; Hindi: movie reviews; Bengali: mixed social media/book reviews). They also differ in class balance, as Marathi's training set is close to balance while Hindi and Bengali are naturally imbalanced. Finally, they differ in training set size, as Marathi has 14,628 training examples, which is over five times more than Hindi's 2,480. Any of these factors could easily

explain the gap rather than any inherent feature of the Marathi language itself (Section 8, Limitations). Fine-tuned mBERT (Section 6.4) widens this gap instead of narrowing it. Marathi's advantage over Hindi increases from about 19.6 points with LR to about 25.7 points with mBERT. This supports a data-scale explanation, as transformers benefit more from larger datasets than linear models using sparse features. Matching the pattern seen for Hindi in [1], the neutral sentiment class has the lowest F1-score across all three languages and all models (LR F1: Hindi 0.4674, Marathi 0.7138, Bengali 0.3958). In Bengali, the drop is severe, trailing roughly 28–30 points behind positive and negative scores. Fine-tuned mBERT partly narrows this gap, raising the Bengali neutral F1-score to approximately 0.49. This gap may partly come from residual uncertainty in the Bengali label mapping (Section 3.1) rather than a purely linguistic effect. Future work should inspect a sample of misclassified neutral Bengali texts to confirm this.

8. Limitations

(i) A single transformer model (mBERT) is fine-tuned per language (Section 6.4). Language-specific pretrained models (MuRIL for Hindi/Marathi, MahaBERT for Marathi, BanglaBERT for Bengali) are not evaluated in this paper. They could plausibly outperform mBERT given their targeted pretraining data. This is left for future work (Section 9). (ii) Each language is modeled independently. No cross-lingual transfer or shared-parameter training is attempted. Therefore, this paper does not address whether sentiment knowledge learned from one Indic language benefits another. That question is left for a planned follow-up study. (iii) The three datasets differ in domain, collection method, and size (see Table 1). Any cross-language performance difference may reflect dataset characteristics as much as language-specific difficulty. It should not be over-interpreted as a claim about the relative computational difficulty of the three languages. Section 6.4 and Section 7 discuss evidence that dataset size and balance, rather than language, likely drive most of the observed gap. (iv) As with [1], all three test sets are of modest-to-moderate size. Results — while cross-validated for the classical models — should be confirmed on larger or additional datasets before strong generalization claims are made. Cross-validation was not performed for the mBERT models due to the computational cost of repeated transformer fine-tuning.

9. Conclusion

This paper expands the Hindi Sentiment Classification Framework (HSCF) from [1] to Marathi and Bengali. We use the exact same pipeline for each language separately. It combines TF-IDF with simple machine learning models—Multinomial Naive Bayes (MNB) and Logistic Regression (LR). We also include a fine-tuned mBERT transformer as a baseline. Logistic Regression reached a test accuracy of 56.13% on Hindi, 75.73% on Marathi, and 62.30% on Bengali. It outperformed Naive Bayes in every language, which scored 52.90%, 75.38%, and 60.66% respectively. These performance differences were statistically significant. Fine-tuned mBERT improved accuracy even further, reaching 56.77% for Hindi, 82.44% for Marathi, and 68.22% for Bengali. Neutral sentiment was the hardest class to identify across all models and languages. This difficulty was most extreme in Bengali. Marathi performed much better overall than Hindi and Bengali. Moving to mBERT made this accuracy gap wider instead of smaller. We suggest this gap comes from dataset size, domain, and class balance rather than actual language differences. Matching dataset size and domain in future studies will help test this idea. For future work, we plan to test language-specific transformers. We will try MuRIL [8] for Hindi and Marathi, IndicBERT [9], and BanglaBERT for Bengali instead of shared mBERT. We also plan to run cross-lingual transfer experiments across all three languages, following work by Pires et al. [10].

References

1. Singh, S., Divya, G., K, R., S, D.K., Vijayan, L., Desai, U.: Automated sentiment analysis of Hindi text using machine learning techniques. In: Proc. 9th Int. Conf. Inventive Computation Technologies (ICICT-2026), pp. 2181–2186. IEEE (2026)
2. Bali, K., Sharma, J., Choudhury, M., Vyas, Y.: "I am borrowing ya mixing?" An analysis of English-Hindi code mixing in Facebook. In: Proc. First Workshop on Computational Approaches to Code Switching, pp. 116–126 (2014)
3. Kumar, R., Gupta, S., Sharma, P.: Comparative analysis of machine learning algorithms for text sentiment classification. J. Big Data 11, 1–20 (2024)
4. Kulkarni, A., Mandhane, M., Likhitkar, M., Kshirsagar, G., Joshi, R.: L3CubeMahaSent: a Marathi tweet-based sentiment analysis dataset. In: Proc. 11th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA), pp. 213–220 (2021)
5. Pingle, A., Vyawahare, A., Joshi, I., Tangsali, R., Joshi, R.: L3Cube-MahaSent-MD: a multi-domain Marathi sentiment analysis dataset and transformer models. arXiv preprint arXiv:2306.13888 (2023)
6. Kabir, M., Mahfuz, O.B., Raiyan, S.R., Mahmud, H., Hasan, M.K.: BanglaBook: a large-scale Bangla dataset for sentiment

analysis from book reviews. In: Findings of the Association for Computational Linguistics: ACL 2023 (2023)
7. Islam, M.E., Chowdhury, L., Khan, F.A., Hossain, S., Hossain, M.S., Rashid, M.M.O., Mohammed, N., Amin, M.R.: SentiGold: a large Bangla gold standard multi-domain sentiment analysis dataset and its evaluation. In: Proc. 29th ACM SIGKDD Conf. Knowledge Discovery and Data Mining, pp. 4207–4218 (2023)
8. Khanuja, S., et al.: MuRIL: multilingual representations for Indian languages. arXiv preprint arXiv:2103.10730 (2021)
9. Doddapaneni, S., Aralikatte, R., Ramesh, G., Goyal, S., Khapra, M.M., Kunchukuttan, A., Kumar, P.: Towards leaving no Indic language behind: building monolingual corpora, benchmark and models for Indic languages. In: Proc. 61st Annual Meeting of the ACL, pp. 12402–12426 (2023)
10. Pires, T., Schlinger, E., Garrette, D.: How multilingual is multilingual BERT? In: Proc. 57th Annual Meeting of the ACL, pp. 4996–5001 (2019)
11. Conneau, A., et al.: Unsupervised cross-lingual representation learning at scale. In: Proc. ACL, pp. 8440–8451 (2020)
12. Ullah, F., et al.: Prompt-based fine-tuning with multilingual transformers for language-independent sentiment analysis. Sci. Rep. 15, 20834 (2025)
13. https://github.com/satyapal2240/1_for_conference_HSCF_A_Lightweight_and.ipynb/blob/main/A_Lightweight_NLP_Framework_for_multilang.ipynb
14 IIT Patna Movie Reviews Hindi Sentimental Analysis Dataset. [Online]. Available: https://www.kaggle.com/code/khushipitroda/iitpatna-movie-reviews-hindi-sentiment-analysis
14 https://github.com/pramanik-souvik/SentNob-Bengali-Sentiment-Analysis.git
15. https://github.com/l3cube-pune/MarathiNLP.git