Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Code-Mixed Corpora for Indian Social Media: Baselines and Insights for Hinglish Language Identification

Kavita Srivastava

kavita332211@gmail.com

IITM

Research Article

Keywords: Low-Resource NLP, Linguistic Diversity, AI for India, Code-Mixed NLP, Hinglish, Language Identification, Indian Languages, and Social Media Text

Posted Date: August 13th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10672730/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Abstract

In general, Indian communities interact in Hinglish, a Hindi and English blend that is frequently written in Roman script so is the Indian social media.On code-mixed text, where spelling variations, frequent language switching, and noisy Romanization are common, multilingual models like mBERT and XLM-R perform poorly, despite their advancements in cross-lingual natural language processing. In this research work, a dataset named COMI-LINGUA has been used which has nearly over 100,000 expert-annotated examples from various NLP tasks, to explore token-level Hinglish Language Identification (LID). In this work, we have proposed a method that provides a lightweight yet efficient baseline by using a straightforward pipeline based on character-level TF-IDF features and Logistic Regression. On the validation set, the model's accuracy is 96.06% with a macro-F1 of 0.9523; on the test set, it achieves 95.92% accuracy with a macro-F1 of 0.9509. also, the error analysis has been done that indicates

persistent problems like short token ambiguity, Romanization-induced spelling variations, and loanwords that conflate languages. This work makes three contributions: (i) a reproducible baseline that performs roughly 7% better than multilingual benchmarks; (ii) an error-driven analysis of the primary difficulties in Hinglish LID; and (iii) a roadmap for expanding the work to multilingual environments, context-aware models, and fairness-focused methodologies. The results provide a useful standard and guidance for inclusive language technologies in India going forward.

1. Introduction

India has population diversity. With 122 scheduled languages and more than 19,000 dialects, India has one of the most linguistically diverse populations in the world (Census of India, 2011). Moreover, language switching is a common practice in this setting and is becoming more noticeable on social media sites like Facebook, YouTube, WhatsApp, and Twitter (Bali et al., 2014). Hinglish, a Hindi-English combination usually typed in Roman script, is basically the most common bilingual communication style in online discourse.

While code-mixed language usage seems normal to humans, it presents significant difficulties for natural language processing (NLP) systems. Multilingual models trained primarily on monolingual data, like mBERT and XLM-R (Devlin et al., 2019; Conneau et al., 2020), suffer from Hinglish-specific problems like frequent language switching, short ambiguous tokens (is, me, ki), and inconsistent spellings (acha, achha), all of which impair LID and downstream task performance (Khanuja et al., 2020).Therefore, addressing this issue requires large-scale Hinglish data and is available through recent resources. For pre-training, the L3Cube-HingCorpus (Nayak & Joshi, 2023) offers more than a billion tokens of raw code-mixed text. In addition, COMI-LINGUA (Sheth et al., 2025) provides over 125,000 annotated examples for five fundamental NLP tasks: sentiment analysis, POS tagging, token-level LID, NER, and translation. We have used in this paper, COMI-LINGUA to perform a thorough error analysis and establish a robust token-level LID baseline. While highlighting long-term issues and potential paths forward for developing

inclusive multilingual language models suited to India's digital environment, pur work also shows that lightweight character-based models can achieve high accuracy.

Motivation and Objectives

In fact, there is huge discrepenciy between how people actullay communicate on social media and how the multilingual models can handle it. This is the motivation of our work. Online, code-mixed Hinglish has become commonplace; however, the majority of large models are unable to handle the irregular spellings, short ambiguous words, and loanwords that are prevalent in Hinglish because they were designed for monolingual or parallel text. The objective of the proposed work is to partially fill this gap by:

1. Using lightweight character-based models to create a trustworthy token-level LID baseline for Hinglish.
2. Using the COMI-LINGUA dataset to illustrate the efficacy of straightforward, domain-aware modeling.
3. Finding and examining common sources of inaccuracy, such as loanwords, ambiguous short tokens, and variations in romanization.
4. Putting Hinglish LID in the forefront of the larger initiative to create inclusive LLMs for India's multilingual online community.

Contributions

Following are the contribution of our work:

1. Using character n-grams with TF-IDF and Logistic Regression, we create a simple yet efficient pipeline that achieves near-state-of-the-art performance (~ 96% accuracy, ~ 95% macro-F1) on COMI-LINGUA. Our baseline is repeatable, interpretable, and performs roughly 7% better than previously published multilingual systems.
2. A thorough error analysis identifies the primary obstacles facing Hinglish LID and makes recommendations for future research, including sequence-aware models, multilingual extensions, federated training, and fairness-aware natural language processing.

In other words, this work advances the long-term objective of developing language technologies that honor and reflect India's complex linguistic reality in addition to addressing the immediate task of identifying Hindi and English tokens in Hinglish text.

2. Related Work

We have used COMI-LINGUA dataset. With over 125,000 instances and over 376,000 annotations, the COMI-LINGUA dataset (Sheth et al., 2025) is the most extensive resource for Hindi–English code-mixed text to date. The main tasks handled by this dataset are – Matrix language identification, token-level LID, part-of-speech tagging, named entity recognition, and machine translation. In fact, 3 bilingual experts

independently annotated each instance. It resulted in high inter-annotator agreement (κ = 0.81) between the Devanagari and Roman scripts. According to evaluations, one-shot prompting increases the efficacy of closed-source large language models in structured tasks, even though they perform well in zero-shot settings. Results are remarkable (98.77 F1 in MLI and 95.25 F1 in NER) from fine-tuned models on COMI-LINGUA. Therefore, the dataset is established as a standard for multitask code-mixed NLP and the development of inclusive language technologies. Furthermore, other datasets are also available. The L3Cube-HingCorpus (Nayak & Joshi, 2023), the first extensive Hinglish dataset gathered from Twitter. No doubt, it is another significant contribution. It contains 1.04 billion tokens spread across 52.9 million Roman-script sentences. It made it possible to create pre-trained models like HingBERT, HingMBERT, HingRoBERTa, and HingGPT, which have proven to perform well on GLUECoS benchmark tasks like sentiment analysis, POS tagging, NER, and LID. Hinglish text is processed in a scalable manner by using other datasets like largest Hinglish LID dataset, HingLID, and a production-grade classifier, HingBERT-LID. The SAIL shared task at ICON-2017 (Patra et al., 2017) was one of the first attempts in the field. It offered baseline systems, evaluation protocols, and annotated Hindi–English and Bengali–English sentiment datasets. Important issues like informal grammar, transliteration inconsistencies, and orthographic variation were revealed by the shared task. The majority of participant systems explored conventional machine learning and feature-based methods, laying the groundwork for later research on sentiment analysis of languages with mixed codes. The field has been controlled by multilingual benchmarks in addition to Hinglish. By providing 11 tasks in various languages that cover both natural language generation and understanding, XGLUE (Liang et al., 2020) expanded the GLUE benchmark to multilingual contexts. The potential and drawbacks of multilingual pre-training were demonstrated by evaluations of models like Unicoder, mBERT, XLM, and XLM-R, solidifying XGLUE as a key resource for research on cross-lingual generalization.

Similarly, one of the earliest systematic studies on word-level language identification in social media texts in Bengali and Hindi was given by Das and Gambäck (2013). They laid the foundation for subsequent token-level LID research with their hybrid system of character n-grams, dictionaries, and SVMs, which obtained ~ 79% F1 and exposed sociolinguistic patterns of code-mixing. With a thorough assessment of multilingual encoders in 40 languages and nine tasks—including classification, structured prediction, and retrieval—the XTREME benchmark broadened this approach (Hu et al., 2020). The findings demonstrated that while models such as mBERT and XLM-R perform at nearly human levels in English, they deteriorate in low-resource languages, especially when it comes to tasks involving a lot of syntax. Since then, XTREME has established itself as a common norm for evaluating the robustness and fairness of multilingual systems. In addition, the IndicNLPSuite brought resources for 11 Indian languages, such as the IndicGLUE benchmark, 8.8B tokens, FastText embeddings, and ALBERT-based models (Kakwani et al., 2020).

IndicGLUE, a crucial element for advancing Indic NLP research, covers a range of tasks such as genre classification, cloze QA, and NLI, and its models consistently outperform earlier baselines. According to Khanuja et al. (2020), GLUECoS provided the first code-switching benchmark, covering six tasks in both English–Hindi and English–Spanish: LID, POS, NER, sentiment, QA, and NLI. In addition to providing a

consistent testbed for the development of code-mixed and multilingual natural language processing, it showed that multilingual transformers outperform cross-lingual baselines and improve significantly when improved on synthetic code-switched data.

Recent studies have also focused on speech. Biswas et al. (2025) modified Whisper large-v2 for Hindi–English code-switched ASR using LLM-driven synthetic data generation and on-the-fly code-mix simulation. In comparison to the baseline MER of roughly 52%, augmentation reduced errors by 31%, proving that synthetic data is a scalable approach for low-resource ASR. Furthermore, the HiACC corpus was the first Hinglish speech dataset to encompass both adults and children, containing 5.24 hours of read and spontaneous speech (Singh et al., 2025).

At about 42% WER, baseline ASR models trained on monolingual data performed poorly, underscoring the challenge of spontaneous Hinglish and addressing a significant research gap for child-inclusive ASR. Innovations such as MaskLID, which successively mask dominant-language attributes to detect secondary languages, provide a lightweight, unsupervised solution for code-switching LID at the methodological level (Kargaran et al., 2024). MaskLID demonstrated the potential of iterative feature masking as a resource-efficient method for intra-sentential LID by improving the recognition of mixed-language sentences when applied to GlotLID and OpenLID.

Research Gap

Still, there is a huge research gap in Hinglish NLP. Early shared workloads like SAIL and FIRE provided modest annotated corpora, but multilingual benchmarks like XGLUE, XTREME, IndicGLUE, and GLUECoS mainly focus on monolingual or synthetic scenarios, leaving genuine code-mixed evaluation understudied. More recent resources such as COMI-LINGUA and L3Cube-HingCorpus offer large-scale corpora and expert annotations, but there aren't many systematic lightweight baselines and error-driven studies available. For multilingual LMs like mBERT and XLM-R, issues like hinglish loanwords, short-token ambiguity, and inconsistent Romanization still exist. Whisper-based adaptations and speech datasets such as HiACC exhibit similar challenges in ASR, despite their limited scope. Techniques like MaskLID have not been extensively tested on Indian corpora, despite their potential. The primary research gap is thus the lack of reproducible baselines and comprehensive error analysis tailored to Hinglish, which are essential for developing inclusive and trustworthy LLMs for India's linguistically diverse environment.

3. Methodology

Building a straightforward but efficient baseline for recognizing Hindi and English tokens in Hinglish text is the aim of proposed work. In order to capture the irregularities of code-mixed data, such as noisy spellings, inconsistent Romanization, and short ambiguous tokens, we design a lightweight pipeline that is easily reproducible on standard machines, as opposed to depending on bulky multilingual models.

3.1 Problem Formulation

Given a sequence of tokens $T = \{t_1, t_2, \ldots, t_n\}$, the task is to assign each token t_i a label from the set $\{HI, EN\}$, representing **Hindi** or **English**. This constitutes a **token-level classification problem**, where tokens are considered independently of sentence context in the baseline system.

3.2 Dataset

The experiments that we hace conducted use the COMI-LINGUA dataset (Sheth et al., 2025). Over 125,000 annotated Hinglish instances from various NLP tasks are included in this dataset. Token-level labels were employed for LID: approximately 0.97 million tokens were used for training, 0.19 million for validation, and 0.19 million for testing. The dataset is a good benchmark for our research because it includes both Hindi and English tokens in Romanized script. Although there are larger unlabeled corpora, like L3Cube-HingCorpus (Nayak & Joshi, 2023), they were not used in these experiments directly; rather, they are mentioned as useful resources for pre-training in the future.

3.3 Feature Engineering

We used character-level features instead of pretrained embeddings, which frequently don't work with noisy Romanized text. English borrowings typically finish in suffixes like "-ing" or "-ed," whereas Hindi words written in Roman script could end in "-aa" or "-wala." Hinglish words frequently include distinctive subword hints. To document such patterns, we extracted character n-grams of sizes 2 to 5. Each token was represented by a TF–IDF weighted sparse vector, which reduces the effect of spelling variation and increases robustness against short or fragmented words.

3.4 Classification Model

We used the Saga solver in conjunction with a Logistic Regression classifier. This model handles high-dimensional sparse input well and is interpretable and computationally efficient. Up to 200 iterations of L2 regularization were used during training, and validation loss was used to track convergence. Hyperparameters were automatically adjusted during training, and the full-batch gradient descent method performed consistently on the token-level TF-IDF vectors.

3.5 Training and Evaluation

Basically, training has been conducted on ~ 0.97M COMI-LINGUA tokens. Then validation is done on ~ 0.19M tokens used for tuning and early stopping. Finally, testing is done on ~ 0.19M tokens. In order to evaluate the performance, we have used three matrices – Accuracy, Macro-F1, and Confusion Matrix. Basically, Accuracy represents the proportion pf correctly classified tokens, whereas, Macro-F1 is the harmonic mean of precision and recall, treating both classes equally. In order to examine common misclassification patterns, Confusion Matrix is used.

3.6 Experimental Pipeline

Figure 1 shows the experimental pipeline. Preprocessing extracts token–label pairs from COMI-LINGUA. Vectorization generates 2–5 character n-grams and compute TF–IDF weights. Classification trains Logistic Regression on the resulting sparse vectors. Evaluation reports Accuracy, Macro-F1, and

confusion matrix results. Error Analysis determines misclassified tokens manually to identify error categories such as short ambiguous words, Romanization variants, and loanwords.

4. Experiments and Results

4.1 Experimental Setup

The experimental setup is shown in Fig. 2. According to the typical split outlined in the dataset, we used around 0.97 million tokens from the COMI-LINGUA dataset for training, 0.19 million for validation, and 0.19 million for testing. Each token in these sets was assigned a binary label that denoted its English or Hindi origin. Inspired by the extremely varied and noisy nature of Romanized Hinglish, we weighted character-level n-grams (n = 2–5) for feature representation using TF–IDF. A Logistic Regression classifier trained with the saga solver was fed the sparse vectors that were produced. Scikit-learn was used for training and evaluation, and a typical CPU environment was used for execution. Due to the system's lightweight and portable design, the entire pipeline can be readily replicated without the need for specialized hardware or GPUs.

4.2 Quantitative Results

Table 2 provides a summary of our baseline model's outcomes. The system's accuracy on the validation set was 96.06%, and its macro-F1 score was 0.9523. Performance was consistent on the held-out test set, with an accuracy of 95.92% and a macro-F of 0.9509. These findings demonstrate that, in spite of the noise and variation common in social media text, character-level features offer adequate discriminative power for Hinglish token-level classification.

Table 1
Performance on validation and test sets

Split	Accuracy	Macro-F1
Validation	96.06%	0.9523
Test	95.92%	0.9509

4.3 Error Analysis

According to the test set's confusion matrix (Fig. 3), the majority of errors fall into three recurrent categories which are discussed below. So, improvement using sequence-level models or domain-adaptive approaches need to be explored to overcome challenge of Hinglish LID.

1. **Short token ambiguity**: Depending on the context, very common words like "is," "me," and "ki" can be classified as either Hindi or English. They frequently deceive the classifier when handled separately.
2. **Variants in romanization**: A single Hindi word, such as acha, achaa, or achha, often appears in multiple spellings. These variations lessen prediction consistency and fragment the feature space.

3. **Loanword misunderstandings**: Hindi sentences frequently contain English borrowings like party or school. Such tokens sometimes cause confusion in the system when embedded in contexts where Hindi is the primary language.

6. Discussion and Conclusion

This study demonstrates the challenges and opportunities in Hinglish Language Identification (LID). With a ~ 96% accuracy and ~ 95% macro-F1 performance, a character n-gram + TF–IDF + Logistic Regression model shows that well-designed character-level features are strong enough to compete with more intricate multilingual systems. The model is a useful starting point for additional research because, in addition to accuracy, it is still effective, interpretable, and reproducible on low-end hardware. Systematic flaws were also exposed by the error analysis. Without context, short tokens like is, me, and ki are challenging to categorize. Variants of romanization, such as acha, achaa, and achha, cause errors and divide the representation space. When loanwords like party and school are used in Hindi sentences, they can also lead to incorrect classifications. Due to their lack of exposure to the phonetic, orthographic, and sociolinguistic subtleties of Hinglish, multilingual LLMs trained on generic corpora frequently perform poorly, as these patterns demonstrate.

The proposed method presents a baseline for Hinglish token-level LID that is repeatable and comprehensible. It provides a proof that lightweight character-level models perform roughly 7% better on COMI-LINGUA than multilingual baselines. Also, an error-driven analysis identifying the primary difficulties with Hinglish LID.

In addition to establishing a strong standard, the study emphasizes the value of inclusive and domain-specific techniques for developing LLMs that accurately reflect India's multilingual ecosystem. Future directions include multilingual extension, context-aware modeling, and fairness-aware federated training to create scalable and equitable NLP systems for diverse language communities.

References

1. Sheth R, Beniwal H, Singh M (2025) COMI-LINGUA: Expert Annotated Large-Scale Dataset for Multitask NLP in Hindi-English Code-Mixing. *arXiv preprint arXiv:2503.21670*
2. Patra BG, Das D, Das A (2018) Sentiment analysis of code-mixed indian languages: An overview of sail_code-mixed shared task@ icon-2017. *arXiv preprint arXiv:1803.06745*
3. Nayak R, Joshi R (2022) L3Cube-HingCorpus and HingBERT: A code mixed Hindi-English dataset and BERT language models. *arXiv preprint arXiv:2204.08398*
4. Kargaran AH, Yvon F, Schütze H (2024) MaskLID: Code-switching language identification through iterative masking. *arXiv preprint arXiv:2406.06263*
5. Singh S, Singh M, Kadyan V (2025) HiACC: Hinglish adult & children code-switched corpus. Data Brief, p.111886

6. Biswas A, Shevelev O, Abdaoui A, Tyagi V, Boumadane A (2025) Adapting Whisper for low-resource Hindi-English Code-Mix speech with on-the-fly Augmentation & LLM-Synthesised Data. In *Proc. Interspeech 2025* (pp. 4293–4297)
7. Khanuja S, Dandapat S, Srinivasan A, Sitaram S, Choudhury M (2020) GLUECoS: An evaluation benchmark for code-switched NLP. *arXiv preprint arXiv:2004.12376*
8. Kakwani D, Kunchukuttan A, Golla S, NC G, Bhattacharyya A, Khapra MM, Kumar P (2020) November. IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *Findings of the association for computational linguistics: EMNLP 2020* (pp. 4948–4961)
9. Hu J, Ruder S, Siddhant A, Neubig G, Firat O, Johnson M (2020) November. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International conference on machine learning* (pp. 4411–4421). PMLR
10. Das A, Gambäck B (2013) Code-mixing in social media text. Traitement Automatique des Langues 54(3):41–64
11. Liang Y, Duan N, Gong Y, Wu N, Guo F, Qi W, Gong M, Shou L, Jiang D, Cao G, Fan X (2020) XGLUE: A new benchmark dataset for cross-lingual pre-training, understanding and generation. *arXiv preprint arXiv:2004.01401*

Figures



Figure 1

Experimental Pipeline

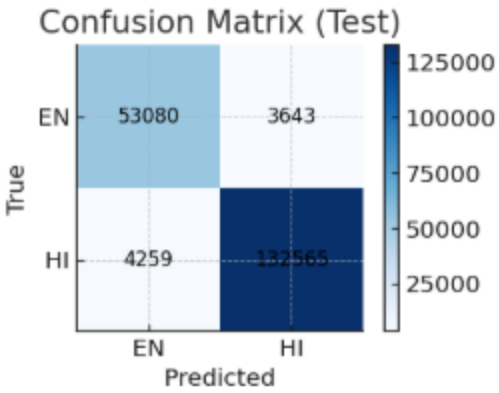


Figure 2

Confusion Matrix

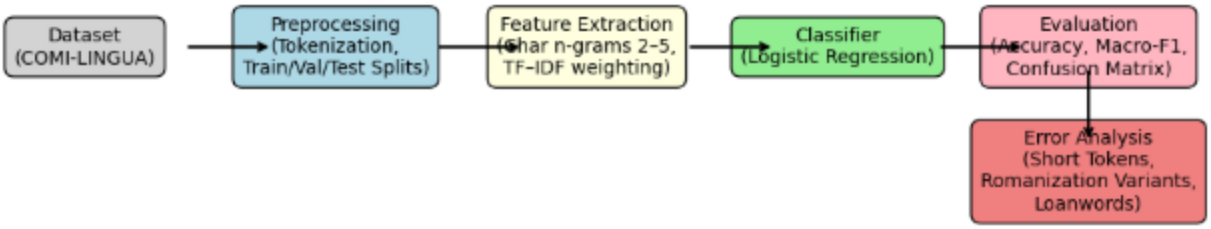


Figure 3

Experimental Setup