

Sentiment Analysis of Code-Mixed Hindi–English Social Media Text Using Transformer Models

Sahil Arora

Dept. of CSE, Your Engineering College, Chandigarh
sahil@youremail.edu

Nidhi Bansal

Dept. of CSE, Your Engineering College, Chandigarh
nidhi@youremail.edu

Abstract— Social media users in India frequently mix Hindi and English in the same sentence, often writing Hindi in Roman script. Such code-mixed text is difficult for sentiment models trained on a single language. This paper evaluates transformer models fine-tuned on a labelled code-mixed corpus and proposes a simple transliteration-aware pre-processing step. Results show that multilingual transformers outperform classical baselines, and that normalising spelling variants improves performance further. Values in this template are sample figures.

Index Terms— *code-mixing, sentiment analysis, Hinglish, transformers, natural language processing*

This is a template: all data, results and references are placeholders for the author to replace.

I. INTRODUCTION

Code-mixing — switching between languages within a sentence — is common in informal communication in multilingual societies. A comment such as “movie was too good, ekdum paisa vasool” combines English and Romanised Hindi, which confuses models trained on either language alone.

Businesses, researchers and public bodies analyse social media sentiment to understand opinions. Accurate analysis of code-mixed text is therefore useful, but the lack of standard spelling and labelled data makes the task challenging.

The main contributions of this paper are as follows:

- A comparison of classical and transformer-based models on code-mixed sentiment data.
- A transliteration-aware normalisation step for common spelling variants.
- An error analysis highlighting sarcasm and spelling as major challenges.

The rest of the paper is organised as follows. Section II reviews related work, Section III describes the proposed fine-tuning approach, Section IV presents the experimental setup, Section V reports and discusses the results, and Section VI concludes the paper.

II. RELATED WORK

Shared tasks on code-mixed sentiment have produced benchmark datasets and baselines [1].

Character-level models handle spelling variation better than word-level models [2].

Multilingual transformers pre-trained on many languages transfer reasonably well to code-mixed text [3].

Normalisation and back-transliteration have been explored to reduce vocabulary sparsity [4].

Ref.	Approach	Data	Limitation
[1]	n-gram + SVM	Shared-task data	Spelling variation
[2]	Char-CNN	Tweets	Short context
[3]	Multilingual BERT	Mixed corpus	Romanised Hindi gaps
[4]	Back-transliteration	Comments	Error propagation

TABLE I. Summary of related work

III. PROPOSED FINE-TUNING APPROACH

Text is first cleaned by removing URLs and user mentions and by expanding common abbreviations. A dictionary of spelling variants maps forms such as “bahut”, “bohot” and “bht” to a single form.

The normalised text is tokenised and passed to a multilingual transformer, which is fine-tuned with a classification head for three classes: positive, negative and neutral.

A. Overview

The pipeline consists of cleaning, variant normalisation, tokenisation, a transformer encoder and a softmax classifier.

Figure placeholder — block diagram of the proposed fine-tuning approach

Fig. 1. Overview of the proposed fine-tuning approach

B. Algorithm

1. Clean the text and remove noise.
2. Normalise spelling variants with the variant dictionary.
3. Tokenise with the model’s tokenizer.
4. Fine-tune the transformer for three-class classification.
5. Evaluate on the held-out test set.

IV. EXPERIMENTAL SETUP

Experiments used a labelled corpus of code-mixed comments (sample size: 15,000; positive 38%, negative 32%, neutral 30%). Models were fine-tuned for four epochs with a learning rate of $2e-5$.

Parameter	Value
Corpus size	15,000 comments (sample)
Classes	3
Max sequence length	128
Learning rate	$2e-5$
Epochs	4
Hardware	Single GPU

TABLE II. Experimental parameters

A. Evaluation Metrics

- Macro F1-score
- Accuracy
- Per-class precision and recall

V. RESULTS AND DISCUSSION

The multilingual transformer with normalisation achieved the best macro F1-score. Classical models struggled with unseen spelling variants.

Model	Accuracy (%)	Macro F1
TF-IDF + SVM	68.2	0.66
Char-CNN	71.5	0.70
Multilingual transformer	78.9	0.78
Transformer + normalisation	81.4	0.80

TABLE III. Comparison with baseline methods (sample data)

Chart placeholder — insert a graph of your results

Fig. 2. Confusion matrix of the best model (sample data)

Most remaining errors involved sarcasm, where positive words express a negative opinion, and very short comments that lack context.

Normalisation helped most for negative comments, which contained more informal spellings.

A. Ablation Study

To understand the contribution of each component, we removed one component at a time and repeated the evaluation. Table IV shows that every component contributes to the final result, with the largest drop observed when the spelling normalisation is removed.

Variant	Macro F1
Full model	0.80
Without spelling normalisation	0.78
Without pre-processing	0.77
Baseline only	0.66

TABLE IV. Ablation results (sample data)

B. Limitations

- Only Hindi–English mixing was studied.
- The variant dictionary was built manually.
- Emojis were removed rather than modelled.

VI. CONCLUSION AND FUTURE WORK

This paper showed that multilingual transformers, combined with simple spelling normalisation, perform well on code-mixed Hindi–English sentiment analysis.

We plan to learn normalisation automatically, model emojis, and extend the work to other Indian language pairs.

Acknowledgment

The authors thank the department and the participants who helped with data collection. Replace this paragraph with your own acknowledgment, including any funding support.

References

- [1] A. Author, B. Author, and C. Author, “Title of the journal article,” *Journal Name*, vol. 10, no. 2, pp. 101–112, 2023.
- [2] D. Author and E. Author, “Title of the conference paper,” in *Proc. Int. Conf. on Topic*, City, 2022, pp. 45–50.
- [3] F. Author, *Title of the Book*, 3rd ed. City: Publisher, 2021.
- [4] G. Author et al., “Title of the article,” *Transactions on Topic*, vol. 7, pp. 220–231, 2024.
- [5] H. Author, “Title of the thesis,” Ph.D. dissertation, Dept. Name, University Name, City, 2020.
- [6] I. Author and J. Author, “Title of the preprint,” *arXiv preprint*, 2024.
- [7] K. Author, “Title of the dataset or software,” Version 2.1, 2023. [Online]. Available: URL
- [8] L. Author, M. Author, “Title of the survey paper,” *ACM Computing Surveys*, vol. 55, 2023.