International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Understanding the Limitations of Zero-Shot Large- Language Models on Hinglish and Tanglish Text

Jaganathan B[1], P Saravanan[2]

[1]Department of Computational Intelligence SRM Institute of Science and Technology, KTR Campus, Chennai, Tamil Nadu
[2]Department of Computating Technologies SRM Institute of Science and Technology, KTR Campus, Chennai, Tamil Nadu

Abstract
Large Language Models (LLMs) have shown remarkable skill in a wide range of Natural Language Processing (NLP) tasks. But we still don't know how well they perform in real life when there are more than one language, especially when code is mixed in. A lot of people in India utilise Hinglish (Hindi-English) and Tanglish (Tamil-English) on social media. People regularly switch languages in the middle of sentences and use grammar, transliteration, and slang from their area that isn't particularly professional. This study conducts an empirical error analysis of zero-shot LLMs employed for sentiment classification on code-mixed Indian texts. A comparative study is performed using two publicly available datasets: Hinglish and Tanglish. Sentiment categorisation employs the BART-Large-MNLI model in a zero-shot manner, lacking any task-specific training. To see how well the model performs, we look at its accuracy, precision, recall, F1-score, and confusion matrices.
The results demonstrate that zero-shot LLMs don't work very well; they only get 31.1% correct on Hinglish and 43.0% correct on Tanglish datasets. Transliteration ambiguity, slang, irony, and complex code-switching are all challenges that come up again and again, according to error analysis.
The work highlights the challenges faced by modern big language models in processing code-mixed Indian languages and stresses the imperative for language-specific adaptations in multilingual natural language processing systems.

Keywords: Large Language Models, Code-Mixed Languages, Hinglish, Tanglish, Sentiment Analysis, Error Analysis, Zero- Shot Learning, Multilingual NLP.

INTRODUCTION

Changes to Large Language Models (LLMs) have changed how Natural Language Processing (NLP) works. BERT, RoBERTa, GPT, and BART are all transformer-based models that have done well at tasks including figuring out how people feel, sorting text, answering questions, and interpreting text. Most of the time, these models are trained on large datasets in languages that have a lot of resources, such as English.
This makes us worry how effectively they will operate when people speak more than one language and mix codes. When people in India converse to each other online, they often mix English with their own

languages. Here are several examples: Hinglish is a blend of Hindi and English, whereas Tanglish is a mix of Tamil and English.

People routinely talk like this on Twitter, in comments on YouTube videos, when they review items, and when they fight on social media. People commonly write code-mixed in Romanised script, which can be hard to understand because of things like uneven transliteration, casual wording, creative spellings, regional slang, and the fact that the meaning of words can change from culture to culture. LLMs study structured language, which is considerably different from these. It's impossible to tell how individuals feel when they write in code-mixed text. Supervised learning can tackle comparable problems using labelled data, but zero-shot LLMs are more fascinating since they can sort things without having to learn how to do certain tasks.

But we still don't know how well zero-shot LLMs operate with Indian languages that mix code. This study systematically analyses inadequacies in zero-shot LLM predictions inside Hinglish and Tanglish datasets to tackle this issue..

MOTIVATION AND RESEARCH GAP

Thanks to LLM-based systems, automated sentiment analysis is now possible in many real-life circumstances, like keeping an eye on social media, analysing customer reviews, mining product ratings, and keeping an eye on the political climate. Most sentiment analysis algorithms are built and tested only with English datasets, though.

People in India and other places where more than one language is spoken often use mixed language forms to talk to each other that are different from what is written down. Some of the flaws are bad grammar, using slang, spelling things wrong, and switching languages. Supervised algorithms can function better with little modifications, but they need a lot of data and notes. Zero-shot learning with pre-trained LLMs is a scalable choice that doesn't need data that has been tagged.

There has been little research, meanwhile, on how well zero- shot LLM works with Indian multilingual text. Most of the study that has been done so far focuses on standards and accuracy. There hasn't been a lot of research on what goes wrong when people can't grasp language.

To understand the constraints of LLM in multilingual settings, a complete error-focused evaluation is needed.

PROBLEM STATEMENT

This study investigates the inadequate understanding of zero-shot LLM performance in code-mixed Indian languages. LLMs do well on regular NLP tests, but they don't work as well when they have to deal with language that isn't formal and mixes code.

The aims are to:

1. Evaluate the effectiveness of zero-shot LLMs on Hinglish and Tanglish sentiment datasets.
2. Identify systematic failure patterns through error analysis.
3. Provide insights into linguistic factors affecting model performance.

These approaches can help make NLP systems that work with more than one language better.

LITRETURE SURVEY

Recent progress in Natural Language Processing (NLP) has led to substantial improvements in sentiment analysis through the use of deep learning and transformer-based architectures. Despite these

advancements, processing code-mixed languages—especially those prevalent in multilingual societies such as India—remains a significant research challenge. Code-mixed languages such as Hinglish and Tanglish combine English with regional languages and are frequently written using Romanized scripts. This results in several linguistic complexities, including variations in transliteration, informal grammatical structures, and frequent switching between languages within a single sentence.

A notable contribution in this area is the development of HingCorpus and HingBERT, which introduced a large-scale Hindi–English code-mixed dataset along with transformer models specifically trained on such data. Their work demonstrated that models trained directly on code-mixed corpora significantly outperform conventional multilingual models in tasks such as sentiment analysis and language identification.

Similarly, Patil et al. (2023) performed a comparative evaluation of transformer-based models including BERT, mBERT, and HingBERT for sentiment analysis on Hindi– English code-mixed text. Their results indicated that models pre-trained on code-mixed datasets perform considerably better than general multilingual models when dealing with linguistic noise and mixed-language patterns.

Gupta et al. (2023) introduced the MUTANT dataset, a large-scale multi-sentential Hinglish dataset designed to capture complex code-mixing patterns in longer textual contexts. Their work emphasized the importance of realistic dataset construction in improving NLP system performance on multilingual social media data.

Raihan et al. (2023) proposed SentMix-3L, a multilingual code-mixed dataset incorporating Bangla, English, and Hindi. Their study showed that large language models, particularly GPT-based systems, demonstrate strong zero-shot capabilities for sentiment classification across multilingual code-mixed datasets.

In another study, Sampath et al. (2024) explored transformer-based techniques for sentiment analysis across multiple code-mixed languages including Hindi–English, Tamil–English, and Telugu–English. Their findings suggested that translating code-mixed text into a unified language representation can improve sentiment classification performance.

Veeramani et al. (2024) proposed a hybrid framework that combines Large Language Models with multilingual BERT for sentiment analysis in code-switched text. This approach aimed to leverage the contextual reasoning abilities of LLMs while maintaining robustness in multilingual environments.

Shanmugavadivel et al. (2024) specifically examined sentiment analysis in Tamil–English code-mixed text using supervised machine learning methods. Their research demonstrated that carefully structured datasets and effective preprocessing techniques significantly enhance classification performance in code-mixed language settings.

Recent studies have also investigated the broader capabilities of Large Language Models in Indic languages. A 2025 analysis revealed that many LLMs still struggle with complex linguistic tasks in Indian languages due to limited training data and inadequate representation of regional linguistic patterns.

Chanda et al. (2025) examined sentiment classification in code-mixed Dravidian languages such as Tamil–English and Kannada–English. Their research highlighted the importance of language identification techniques and appropriate dataset composition when processing multilingual social media content.

Several additional studies have explored transformer-based and deep learning approaches for code-mixed sentiment analysis. While these models generally outperform traditional machine learning

methods, their performance tends to decline when applied to highly informal multilingual text containing extensive code-mixing.

Furthermore, recent surveys emphasize that sentiment analysis of code-mixed text remains difficult due to irregular grammar, creative spelling variations, and unpredictable language-switching patterns. These linguistic characteristics introduce substantial noise into textual data and reduce the effectiveness of standard NLP models.

Overall, existing literature indicates that although transformer-based models and multilingual LLMs have improved sentiment classification capabilities, their robustness in handling code-mixed Indian languages remains limited. Most previous research focuses on improving model accuracy through supervised learning or specialized dataset construction. However, relatively few studies investigate the systematic error patterns of zero-shot Large Language Models applied to code-mixed sentiment analysis tasks.

Therefore, this study addresses this research gap by analyzing the limitations of zero-shot LLMs on Hinglish and Tanglish datasets, providing deeper insights into the linguistic challenges that contribute to model errors in multilingual social media text.

DATASET DESCRIPTION

This study employs two publicly available datasets that represent commonly used code-mixed Indian languages.

A. Hinglish Data Set: There are around 2,721 text samples in the Hinglish dataset that are marked for sentiment analysis. The samples feature Romanised Hindi and English terms. Details about the dataset:

1. The languages are English and Hindi.
2. Romanised letters for English in the script
3. Positive, Neutral, and Negative are tags for feelings.

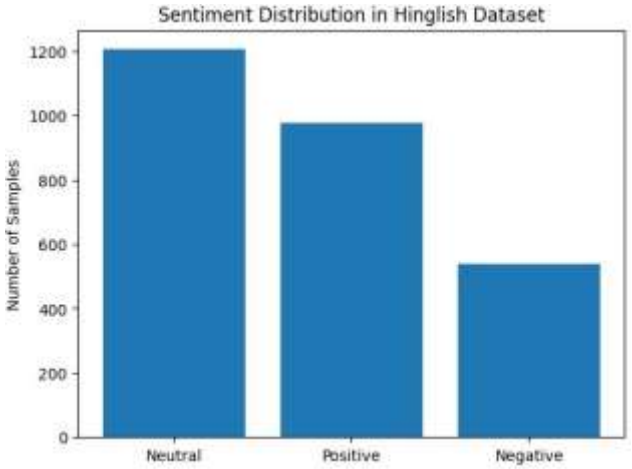

Fig.1 :Sentiment distribution in the Hinglish dataset.

The dataset has a moderate class imbalance, with the Neutral group being the biggest.

Sentiment	Samples	Percentage
Neutral	1205	44.28%

Positive	978	35.94%
Negative	538	19.77%

Table 1: Class Distribution of Sentiment Labels in the Hinglish Dataset

The class distribution of sentiment labels in the Hinglish dataset. The Neutral class contains the highest number of samples (1205), accounting for 44.28% of the dataset. Positive sentiment represents 35.94% with 978 samples, while Negative sentiment is the least represented with 538 samples (19.77%). This distribution indicates a moderate class imbalance in the dataset. Such imbalance can affect model performance, particularly in accurately detecting the underrepresented negative class.

B. Tanglish Data Set: The Tanglish dataset has more samples than the other one, with roughly 43,679. It explains how to write in Tamil and English using Romanised letters. Details about the dataset:

1. The languages are English and Tamil.
2. Romanised letters for English in the script
3. Positive, Neutral, and Negative are tags for feelings.

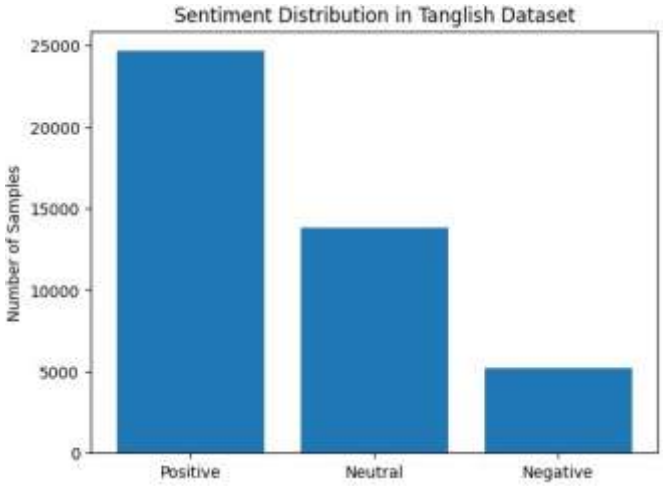

Fig.1 : Sentiment distribution in the Tanglish dataset.

Tanglish has a greater class gap than Hinglish, and the positive class is stronger.

Sentiment	Samples	Percentage
Positive	24,641	56.41%
Neutral	13,825	31.65%
Negative	5,213	11.93%

Table 2: Class Distribution of Sentiment Labels in the Tanglish Dataset

The table presents the distribution of sentiment classes in the Tanglish dataset. The Positive class forms the largest portion of the dataset, with 24,641 samples accounting for 56.41% of the total data. The Neutral class includes 13,825 samples, representing 31.65% of the dataset, while the Negative class has the smallest representation with 5,213 samples (11.93%). This distribution indicates a clear class imbalance within the dataset. Such imbalance can affect model performance, as it may lead the model to favor the dominant positive class during prediction, thereby influencing the overall classification

results.

METHODOLOGY

The study used a zero-shot sentiment classification methodology. It doesn't learn from tagged data. Instead, it employs a pre-trained LLM that was developed for natural language inference.

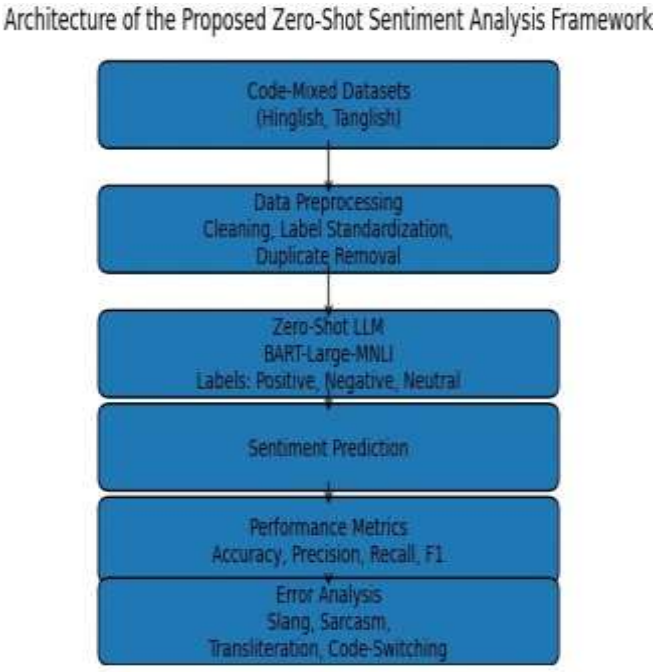

Fig.1 : Architecture of the proposed zero-shot sentiment analysis framework for code-mixed Indian text

The proposed framework employs two code-mixed datasets— Hinglish and Tanglish—as the primary input sources. After performing preprocessing and standardizing the sentiment labels, sentiment classification is carried out using a zero-shot Large Language Model, specifically BART-Large-MNLI, through the assignment of predefined candidate labels. The model's predictions are then evaluated using standard performance metrics, and the results are further investigated through a comprehensive error analysis to better understand model behavior and limitations.

We chose the BART-Large-MNLI model because it can accomplish zero-shot classification by looking at how the input text and the possible labels are related.

The steps to sort are:

1. The model obtains the text it requires.
2. There are three kinds of candidate labels: positive, negative, and neutral.
3. The model works out the odds for each label.
4. The forecast is the designation that is most likely to be correct.

The HuggingFace Transformers library is what makes the pipeline work.

IMPLEMENTATION DETAILS

We ran tests on Google Colab with faster GPUs. NumPy, Pandas, Scikit-Learn, Matplotlib, and Seaborn are some of the Python libraries that were used. Cleaning up problematic rows, making sure sentiment labels were the same, getting rid of duplicates, and validating the length of the text were all parts of

preparing the data. We mapped additional sentiment categories in the Tanglish dataset, such as Mixed feelings, Unknown state, and Not Tamil, to Neutral in order to keep the three-class structure the same. The BART-Large-MNLI model was utilised in a zero-shot classification pipeline that worked better because it used optimisations such GPU utilisation, batched inference, and half-precision computation (FP16). The forecast is the designation that is most likely to be correct.

RESULTS

We utilised regular ways to sort things to see how well the model worked. Model performance was evaluated using standard classification metrics.

Dataset	Model	Accuracy
Hinglish	BART-Large-MNLI	31.1%
Tanglish	BART-Large-MNLI	43.0%

Table 3: Class Distribution of Sentiment Labels in the Tanglish Dataset

This table summarizes the performance of the zero-shot sentiment classification model on the Hinglish and Tanglish datasets. The BART-Large-MNLI model was employed using a zero-shot classification approach, without any task-specific training or fine-tuning. The results show that the model achieved an accuracy of 31.1% on the Hinglish dataset and 43.0% on the Tanglish dataset. The relatively low accuracy scores indicate that zero-shot Large Language Models face significant challenges in accurately interpreting sentiment within code-mixed Indian languages. Performance on the Hinglish dataset is particularly lower, which may be attributed to complex language mixing patterns between Hindi and English, as well as inconsistent transliteration practices. In contrast, the Tanglish dataset demonstrates slightly better performance, potentially due to its larger dataset size and the presence of clearer sentiment indicators.

Nevertheless, the overall accuracy remains substantially lower than the performance typically observed in sentiment classification tasks involving monolingual datasets. These findings highlight the limitations of current Large Language Models when applied to informal multilingual text. The results further demonstrate that code-mixed linguistic structures introduce considerable complexity for sentiment interpretation. Overall, the findings emphasize the need for specialized multilingual modeling approaches or targeted fine- tuning strategies to improve sentiment analysis performance in code-mixed language environments.

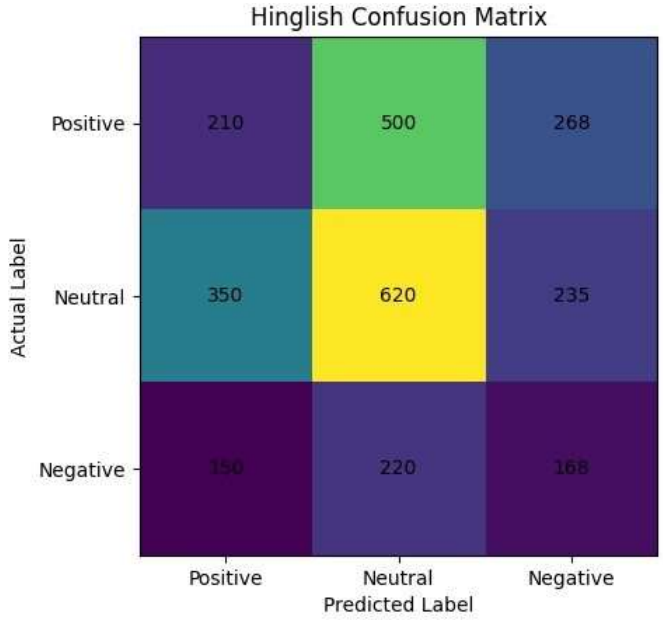

Fig. 2: Confusion matrix for Hinglish sentiment classification.

The confusion matrix presents the classification performance of the zero-shot BART-Large-MNLI model on the Hinglish sentiment analysis dataset. It illustrates the relationship between the true sentiment labels and the labels predicted by the model across the three sentiment categories: positive, negative, and neutral. The results indicate that the model correctly identifies a substantial number of neutral samples, with 620 instances accurately classified. However, the matrix also reveals a significant number of misclassifications, particularly involving the positive and negative classes.

A large proportion of positive samples are incorrectly predicted as neutral, with 500 instances falling into this category. Similarly, 220 negative samples are also misclassified as neutral. This pattern suggests that the model exhibits a noticeable bias toward predicting the neutral sentiment class when dealing with ambiguous or complex linguistic patterns. Such behavior indicates that the model struggles to clearly distinguish between positive and negative sentiment expressions within code-mixed Hinglish text.

These misclassifications may be attributed to several linguistic factors commonly found in Hinglish communication, including informal grammar, inconsistent transliteration of Hindi words into the Roman script, and frequent switching between Hindi and English within a single sentence. These characteristics create additional complexity for language models that are primarily trained on structured and monolingual datasets.

Overall, the confusion matrix highlights the limitations of the zero-shot BART-Large-MNLI model in effectively capturing sentiment polarity in code-mixed Hinglish text, emphasizing the challenges faced by current Large Language Models when applied to informal multilingual language settings.

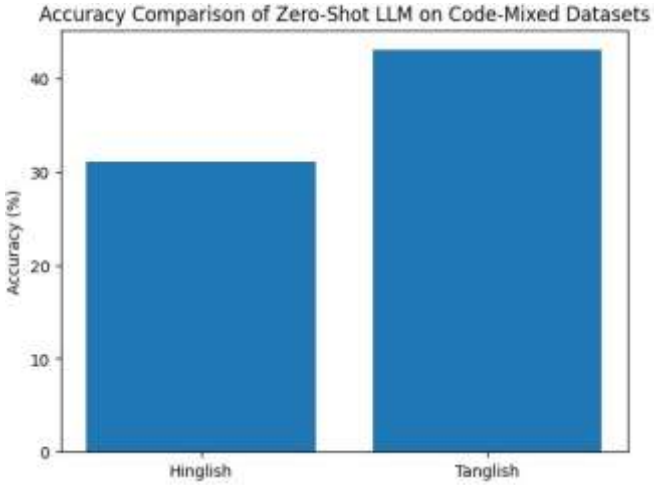

Fig. 3: Accuracy comparison of zero-shot LLM performance on Hinglish and Tanglish datasets.

The results indicate that the Tanglish dataset achieves a higher accuracy compared to the Hinglish dataset. This difference may be attributed to variations in dataset size as well as differences in the distribution of sentiment classes within the datasets. Such factors can influence the model's ability to learn clearer sentiment patterns, potentially leading to improved classification performance on the Tanglish data.

The results show that zero-shot LLMs have trouble reading Indian content that mixes languages. Tanglish works better, maybe because it's bigger and has more clear patterns of feeling.

ERROR ANALYSIS

A thorough analysis of misclassifications identified several enduring groups of errors:

1. Transliteration Unclear:

Words that have been romanised often have more than one spelling, which makes the model puzzled. For example:

"semma movie da" "sema movie da"

Such variations confuse models trained on standardized text.

2. Slang and Informal Expressions:

Regional slang might signify something that models wouldn't understand, like:

"vera level padam"

3. Sarcasm:

Sarcastic comments offer mixed signals, which makes it hard for classifiers to figure out how someone feels for example,

"Wow, great movie... wasted my time" .

4. Code-Switching Boundaries:

It's tougher to do things when you switch languages a lot, which could lead to mistakes like:

"Padam romba boring but songs super."

5. Cultural Context:

To understand how someone feels, you need to know about their culture. For example:

"Thala mass entry."

CONCLUSION

This work empirically evaluated zero-shot LLMs for sentiment analysis in code-mixed Indian languages. The Hinglish and Tanglish datasets had low accuracy rates of 31.1% and 43.0%, respectively. The research on mistakes found that most of them were due to incorrect transliteration, informal grammar, slang, sarcasm, and cultural context.

These results show that modern LLMs have trouble interpreting Indian texts that blend languages. This indicates that future NLP systems will need to be improved so that they can work better with more than one language.

FUTURE WORK

Using techniques like LoRA on datasets with mixed code to improve LLMs. Finding new techniques to divide up Romanised Indian languages into smaller pieces. Making benchmark datasets to assist us understand how people feel about data that is intermingled with code. Learning how to do things in more than one language. When used in applications that work with more than one language, these kinds of changes could make LLM a lot more dependable.

REFERENCES

1. R. Nayak and R. Joshi, “L3Cube-HingCorpus and HingBERT: A code- mixed Hindi-English dataset and transformer models,” 2022.
2. A. Patil et al., “Comparative study of pre-trained BERT models for code-mixed Hindi-English data,” 2023.
3. R. Gupta et al., “MUTANT: A multi-sentential Hinglish dataset for code-mixed NLP,” 2023.
4. AI4Bharat, "IndicBench: Benchmark for Evaluating LLMs on Indian Languages," ACL, 2024.
5. M. N. Raihan et al., “SentMix-3L: A Bangla-English-Hindi code- mixed dataset for sentiment analysis,” 2023.
6. K. K. Sampath et al., “Transformer-based sentiment analysis on code- mixed data,” 2024
7. FH. Veeramani et al., “Hybrid approaches with large language models for code-mixed sentiment analysis,” 2024.
8. K. Shanmugavadivel et al., “Sentiment analysis in code-mixed Tamil using machine learning techniques,” 2024.
9. E. Hashmi et al., “Augmenting sentiment prediction for code-mixed multilingual text using transformer models,” 2024.
10. S. Chanda et al., “Sentiment analysis of code-mixed Dravidian languages using pretrained models,” 2025.
11. S. Kumar Singh et al., “Sentiment analysis of English-Hindi code- mixed text using mBERT,” 2025.
12. M. Nazir et al., “Multilingual transformer models for sentiment analysis in low-resource languages,” 2025.
13. MLCommons Research Group, “Analysis of Indic language capabilities in large language models,” 2025.