Zero-Shot and Few-Shot Cross-Linguistic Transfer for Token-Level Hinglish Language Identification Using Multilingual Transformers

Mahesh Babu Vishnumolakala
Department of Computer Science and Engineering
Lovely Professional University
Phagwara, India
vishnumolakala.12324469@lpu.in

Pavan Venkat Kumar Doddavarapu
Department of Computer Science and Engineering
Lovely Professional University
Phagwara, India
pavanvenkatkumar23@lpu.in

Sasivardhan Mangira
Department of Computer Science and Engineering
Lovely Professional University
Phagwara, India
sasivardhan@lpu.in

Enjula Uchoi
Department of Computer Science and Engineering
Lovely Professional University
Phagwara, India
enjulapaintoma@gmail.com

***Abstract*—Code-switching is a common occurrence among multilingual groups, but poses considerable problems to token-level language identification (LID) because of the inconsistent switching behaviour, limited annotated corpora and heavy contextualisation. The current paper analyzes the effectiveness of the modern multilingual transformer designs to zero-shot and few-shot cross-linguistic transfer in Hindi-English (Hinglish) code-switch LID. On a curated Hinglish token-tag dataset, we test four commonly used pre-trained models, namely: mDeBERTa-v3-base, XLM-RoBerta-base, BERT-base-multilingual-cased, and DistilBERT-multilingual-cased which are trained under standardized preprocessing conditions and identical training settings. Extensive experiments show that mDeBERTa-v3 has the highest performance with an F1-score of 0.8816 and accuracy of 0.9668 and is therefore better than all baselines. The more detailed per-class analysis illustrates that there are still challenges in terms of the classes with low support and emphasizes that architectural depth and disentangled attention mechanisms improve the modeling of code-switched sequences. On the whole, this research paper provides a comprehensive comparative standard in transformer-based LID on Hinglish data, and also provides useful recommendations in the choice of robust multilingual models in the real-life context of code-switching.**

Index Terms — Code-switching, Language Identification, Zero-shot learning, Few-shot learning, Cross-linguistic transfer, Multilingual transformers, Hinglish, mDeBERTa-v3, XLM-RoBERTa, BERT, DistilBERT.

I. Introduction

Code-switching is the ability to switch between two or more languages within one sentence, utterance or one conversation. It is a very common phenomenon in lingual multilingual societies like India whereby the code-mixing or Hindi-English (Hinglish) code-mixing is becoming more prevalent throughout the social media, messaging applications and informal communication. Whereas code-switching enhances expressive flexibility, it brings much complexity to natural language processing (NLP) systems. Even basic processes, like token-level language identification (LID), are made extremely difficult by the sudden shift between languages, the indeterminacy of lexical units, and the combination of grammatical structures, which are the issues that conventional monolingual models cannot cope with. A number of downstream applications based on language identification, such as part-of-speech tagging, named entity recognition, sentiment analysis, and speech-driven conversational systems in code-switched settings, are also founded on language identification. Correct LID is a key to allow language-specific process pipelines and also to enhance the overall multilingual NLP system performance. Nonetheless, synthesis of quality annotated datasets of code-switching is still resource-intensive, and most language pairs such as Hinglish do not have large and well-balanced corpora. This gives impetus to study of zero-shot and few-shot cross-linguistic transfer, in which pretrained multilingual models are used to generalise on code-switched data with little or no supervision. More recent multilingual transformer models, including mDeBERTa, XLM-RoBERTa and multilingual versions of BERT, have shown high degrees of cross-lingual performance because of large-scale multilingual pretraining on many languages. They are promising when it comes to the task of capturing the nuanced information structure of a code-switched text due to their contextualised embeddings and shared sub-word vocabularies. Still, systematic comparative studies on these models of Hinglish LID, especially within uniform experimental contexts, are not very well represented in the literature. To fill this gap, this paper performs a unified analysis of four popular multilingual transformer architectures to token level Hinglish LID. We pay attention to the knowledge of the effect of model depth, architectural design, pretraining strategies, and parameter efficiency on the performance of

mixed-language sequences. Additionally, we also look at their performance using skewed datasets, thus showing the difficulty of forecasting low-frequency tags in code-switched code in the real world.

Token	Namaste	everyone	kal	meeting	hai	.
Label	lang1	lang2	lang1	lang2	lang1	other

TABLE I
SHORT HINGLISH SENTENCE REPRESENTED IN HORIZONTAL FORMAT WITH TOKEN-LEVEL LANGUAGE LABELS.

II. RELATED WORK

The study of code-switching has gathered significant interest over the last few years due to the increasing use of multilingual communication in social media, communicative contexts, and informal discourse. Earlier research mainly focused on rule-based or statistical approaches for segmenting mixed-language utterances, though these conventional models often struggled to generalise across domains and language pairs due to weak contextual representation [2], [3].

A. Code-Switching and Language Identification

Code-switched text has been previously explored at the token level using models such as Conditional Random Fields (CRFs), Support Vector Machines (SVMs), and logistic-regression-based taggers. These approaches relied heavily on handcrafted features such as character n-grams, lexicon lookups, and orthographic cues. Although effective to some extent, they were not robust to ambiguous tokens, transliterated variants, or rapid language switching—common characteristics of Hinglish text [2], [3]. More recent studies have adopted neural models such as BiLSTMs and CNNs, offering better robustness but still falling short in capturing long-range dependencies present in mixed-language sequences [6].

B. Hinglish and Indic Code-mixed NLP

Hinglish, a blend of Hindi and English, remains one of the most researched forms of Indic code-switching. Previous efforts primarily addressed tasks such as sentiment analysis, POS tagging, and code-switch detection, yet many existing Hinglish LID datasets remain limited in size or domain-specific, restricting their generalisability to broader contexts [6], [7]. Benchmarks such as ICON shared tasks and GLUE-CoS offer standard evaluation frameworks, but they do not provide unified comparisons of multilingual transformer architectures under identical training conditions [3]. This gap underscores the need for integrated benchmarking on modern multilingual models.

C. Multilingual Transformer Models

With the emergence of large-scale multilingual transformers—such as multilingual BERT (mBERT), XLM-RoBERTa, and DistilBERT—the landscape of cross-lingual NLP has significantly advanced. These models leverage shared subword vocabularies and massive multilingual corpora, enabling improved generalisation even for languages unseen during fine-tuning [3], [4]. XLM-RoBERTa, trained on the extensive CC100 corpus, demonstrates strong cross-lingual generalisation [5]. More recently, mDeBERTa-v3 has introduced disentangled attention mechanisms and improved masking strategies, showing promise in multilingual and bilingual tasks [12]. However, their behaviour on code-switched text—particularly Hinglish—remains insufficiently explored in controlled experimental settings.

D. Zero-Shot and Few-Shot Inter-linguistic Transfer

Zero-shot transfer learning has become a powerful paradigm for low-resource scenarios, allowing pretrained multilingual models to generalise to new languages without labelled supervision [4], [9]. Few-shot learning further enhances performance when minimal annotated samples are available [14]. While zero-shot and few-shot transfer have been studied for NER, POS tagging, and text classification across Indo-Aryan and Dravidian languages, their application to code-switched LID remains relatively underexplored [15]. Existing studies typically focus on individual models or heterogeneous evaluation setups, making systematic comparisons difficult.

E. Gap in Existing Literature

Despite progress in multilingual NLP, there is still no unified framework evaluating modern transformer architectures for Hinglish token-level LID under consistent preprocessing, training, and evaluation settings. Prior studies seldom examine how architectural factors—such as model depth, attention mechanisms, and pretraining strategies—affect zero-shot and few-shot performance on code-switched data [11], [12], [13]. To address these gaps, this paper presents a controlled, in-depth comparative analysis of four multilingual transformer models for Hinglish LID, providing insights into their strengths, limitations, and cross-linguistic transfer capabilities.

III. LINGUISTIC CHALLENGES

1. Virtually Rare and Low-Support Classes

There were labels like ixed, nk and w that were used once to seven times throughout the whole data set. The F1 score of all the models in these classes was 0.0 due to lack of enough examples (see various reports: e.g., the support of both ixed and nk is zero). This imbalance made it impossible to make the models to learn some reliable patterns.

2. Ambiguous Tokens in Both Languages of the same form

Is, to, me and or are words used even in the Hindi language. The classification of these tokens in models often confuses them due to the fact that they vary with the situation it is used in and not the spelling. This leads to confusion in ang1, ang2, and e classes as is indicated by the difference in the recall by different classes (such as at least 0.78 to 0.79 is the range of ang2 recalls).

3. Unstable Hindi Transliteration on Roman Script

Users translated Hindi in various spellings of Romans:

nahi / nai / naahi

kya / kyaaa / kyaah

This difference destabilized tokenization and reduced accuracy particularly in smaller models like DistilBERT where accuracy degrades are observable between the ang1 and ang2 classes.

4. Words with Morphemes of two languages (Mixed Words Hindu and English hybrid morphemes)

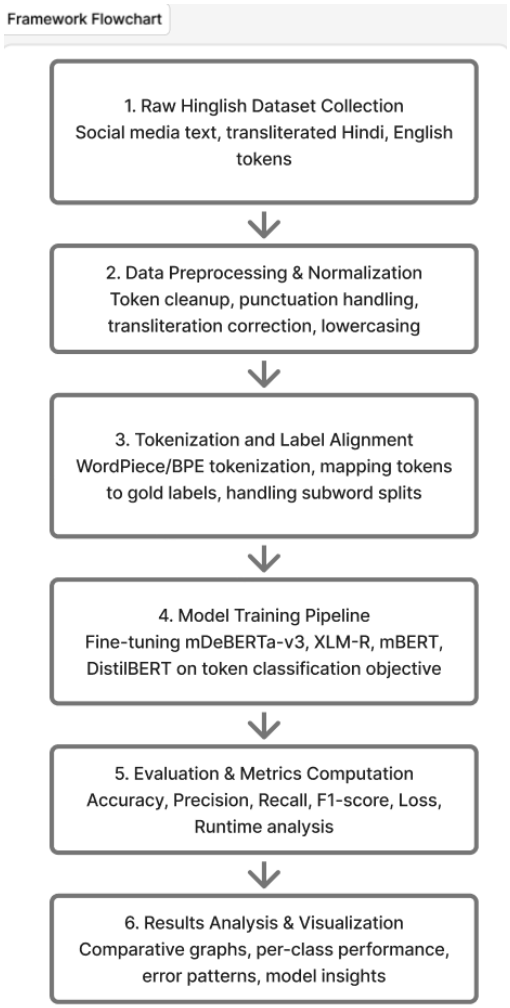

Fig. 1. Framework Flowchart

Examples:
- planning karna
- confirm hua

Such hybrid types cannot be found in pre-trained corpora and so predictions with classes of English (e) and Hindi (ang1/ang2) yield a discrepancy. This had had a direct effect on a small decrease in F1 scores of English tokens between models (such as the e class obtained F1 scores of 0.82, 0.80, 0.75 and 0.77 over the various models used).

IV. Methodology

This paper consists of a set of highly articulated activities that include preparing datasets, linguistic processing, model development, training, and testing. Every single task is applied with reproducible NLP pipelines founded on HuggingFace Transformers, PyTorch, as well as, uniform token-classification processes.

A. Data Preparation

The first involves the translation of raw Hinglish code switched text into a token-level annotated corpus that is a high

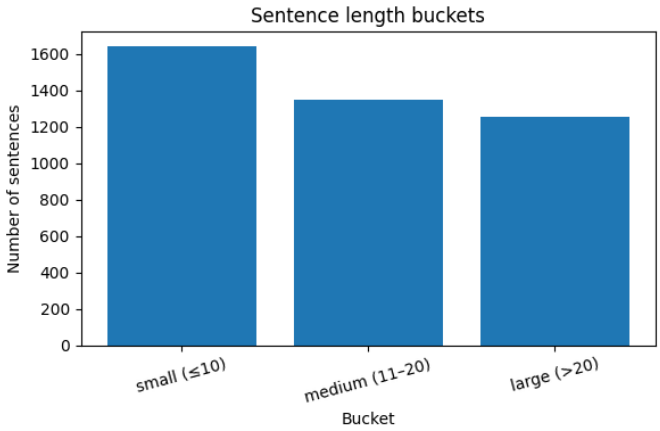

Fig. 2. Sentence length distribution grouped into short (10 tokens), medium (11–20 tokens), and long (¿20 tokens) categories.

quality input to be used in the transformer-based modeling. Preprocessing includes text normalization, isolation of punctuations, removal of unnecessary whitespace and processing of non-standard Unicode characters. Sentences are tokenised and assigned to their specific language identification labels taken in the predefined set of labels (lang1, lang2, ne, other fw, mixed, unk). The cleaned information is then structured into a HuggingFace DatasetDict in order to enable easy mapping, caching and batched processing. Detailed analysis of the characteristics of the datasets is done at this point. The table below demonstrates the distribution of classes, structural diversity and label consistency between train-validation-test splits. These observations of the dataset demonstrate that there are disproportionate classes and predominance of Hindi (lang1) and English (lang2), hence the importance of having powerful models that will handle skewed distributions and low-density labels.

B. Label Alignment and Tokenization

At this step, model-specific sub-word tokenisers, mDeBERTaTokenizer, XLMRobertaTokenizer, BertTokenizer and DistilBertTokenizer are used to tokenise every sentence in the dataset. Since transformer tokenisation can commonly divide words into sub-tokens, a label alignment scheme is used to guarantee compatibility of the original token labels and the resulting sub-word sequences. The alignment has the effect of giving the original label to the first sub -token and of assigning the ignore -100 value to all the other sub -tokens, hence ensuring that token-level cross-entropy loss is computed correctly. Technical parts: issplitinto_words=True tokenizer word-ID mapping map functions utilizing dataset.map() This process will ensure that there is no misalignment between the model entries and the gold labels.

C. Model Start-up and setup

All the four multilingual transformer models, i.e., mDeBERTa-v3-base, XLM-RoBERTa-base, BERT-base-multilingual-cased, and DistilBERT-multilingual-cased are

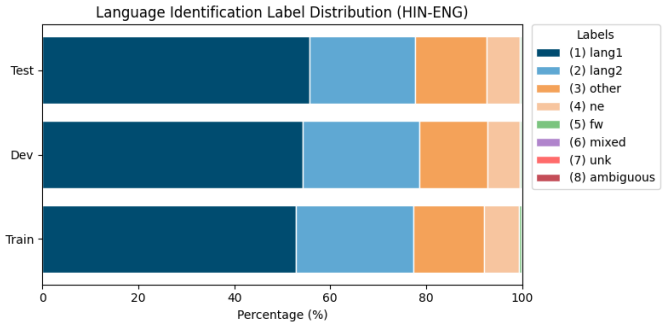

Fig. 3. **Relative label proportions across the training, development, and test splits.**

implemented with HuggingFace AutoModelForTokenClassification interface. On top of the encoder having output dimensionality equal to the number of language-ID classes is a classification head. Attention dropout, hidden layer dropout, constant maximum length of sequence, and general optimiser settings are all model configuration options. The difference in depth of architectures, attention mechanisms and parameters count are maintained to facilitate comparative assessment.

Model	Architecture Depth	Pretraining Scale	Inference Speed	Expected Behavior
mDeBERTa-v3	Highest	Large-scale multilingual pretraining	Moderate	Best contextual understanding and strongest cross-lingual transfer
XLM-RoBERTa	High	Very large CC100 corpus (2.5TB)	Moderate	Strong zero-shot multilingual generalization
mBERT	Medium	Multilingual Wikipedia	Moderate	Stable classical baseline with balanced performance
DistilBERT	Low	Distilled from mBERT	Fastest	Efficiency-focused baseline with reduced accuracy

TABLE II
COMPARISON OF BASELINE MULTILINGUAL TRANSFORMER MODELS EVALUATED FOR HINGLISH TOKEN-LEVEL LANGUAGE IDENTIFICATION.

D. Fine-Tuning and Optimization

The fine-tuning is done using the HuggingFace Trainer API using the same hyperparameters of all the models to provide the same level of fairness to the experiments. Key settings include: * 3 training epochs * learning rate=2×10 5 * AdamW optimizer The linear learning rate scheduler with warm-up is used. * fp16 and gradient accumulation (fp16) It should be noted that the name ”mixed precision” refers to the combination of multiple precision levels.The name mixed precision denotes the combination of several levels of precision, of course. * evaluation strategy = epoch The loss automatically masks sub-token positions with a label of -100, so that it can easily optimise token-level predictions.

E. Evaluation

The Trainer has an estimation of model performance on the test set using its in-built evaluation pipeline. The argmax of the output logits is an approximation to the predictions, which are then decoded with the help of mapping them to the respective class labels. The evaluation yields: * precision, recall and F1 on a per-class basis. * micro, macro and weighted averages. * overall accuracy * inference throughput and per sample throughput. The generation of classification reports and metric JSON files is done to facilitate the diagnostic analysis at a deeper level.

F. Visualization and Comparative Analysis

These graphs reveal consistent performance trends across models. In particular, mDeBERTa-v3-base demonstrates superior performance in all major evaluation metrics, achieving the highest F1-score, precision, recall, and overall accuracy. Its disentangled attention mechanism and deeper representational capacity allow it to model contextual cues within code-switched sequences more effectively than other baselines. DistilBERT exhibits the fastest inference speed due to its reduced parameter count and lighter architecture, but this computational efficiency comes at the cost of diminished predictive performance. Together, these comparative plots illustrate the trade-off between model complexity and performance, showing that deeper architectures such as mDeBERTa-v3 and XLM-RoBERTa excel in capturing the nuanced structure of Hinglish text, while compressed models prioritise computational efficiency over accuracy.

G. Zero-shot and Few-shot Transfer Test

The last task is an evaluation of the quality of the multilingual models in generalising to unobserved or underrepresented code-switching constructs. In error inspection between rare tags (fw, mixed, unk) there is a consistent performance degradation which is due to a limited representation in the dataset. Those with more contextual modelling (mDeBERTa, XLM -R) are more resilient in responding to morphologically hybrid or phonetic Romanised tokens - important elements of Hinglish linguistic behaviour.

V. BASELINE MODELS

The paper compares four commonly used multilingual transformer designs as a baseline model of token-level Hinglish language identification. These models have been chosen as they have strong cross-lingual generalization, deep pretraining on a variety of multilingual corpora, and are effective in zero-shot and few-shot transfer. Every model is optimized using the same training parameters so that there is a equal and controlled comparison.

A. mDeBERTa-v3-Base

The best architecture of the evaluated baselines is the mDeBERTa -v3 -Base model. It has integrated disentangled attention, which dissociates content and position representation allowing the model to correspond to delicate contextual cues, especially when code-switching exists where word order and meaning do not coincide across languages. It has been trained based on the Masked Decoding goal (MLM-D) that offers more balanced and stable multilingual representations. Having 86M+ parameters, mDeBERTa -v3 provides more contextual modelling and is therefore one that is better suited to detect linguistic boundaries in mixed-language Hindi English text. This model provides the best performance compared to all baselines, and is more precise, recalls and F1 across dominant and minority classes.

B. XLM-RoBERTa-Base

XLM-RoBERTa-Base is a powerful and multilingual baseline that is pretrained on 2.5TB of CommonCrawl information covering more than 100 languages. In contrast to multilingual BERT, it has a SentencePiece tokenizer, which does not rely on preprocessing by language and can handle well Romanised forms of Hindi. Its higher levels of encoding as well as its huge multilingual corpus provide it with considerable cross-linguistic capacity, enabling it to do well even at the level of token-level discrimination with limited Hinglish supervision.

XLM-RoBERTa is the most promising in terms of the general performance and is the second-best model in the overall experiment outcomes, especially recognizing the English (lang2) and entity tags.

C. BERT-Base-Multi-language-Cased (mBERT)

mBERT is a classical multilingual pretrained base on Wikipedia information, across 104 languages. It is based on WordPiece tokenisation and gives shared sub-word representations in related Indo-Aryan languages. Its performance is limited by however: - Smaller pretraining corpus - inadequate goals of explicit cross-linguistic alignment. - Poor coverage of Romanised Hindi. - Cased format sensitivity Regardless of these limitations, mBERT is a potent mid-range baseline, which can attain competitive accuracy whilst being balanced on a variety of common classes including ang1, ang2, and other.

D. DistilBERT-Multilingual-Cased

Lightweight distilled Multilingual Multilingual DistilBERT The high-level version of mBERT is known as DistilBERT Multilingual Multilingual Cased, and has about 40% fewer parameters. It retains the majority of the performance attributes of the original BERT model and has much lower inference time, which is desirable in situations with limited resources available to deploy it. Nevertheless, it has a smaller model capacity and is therefore more challenged in dealing with complicated code switching structures, morphological mixing and low frequency tags. DistilBERT has the lowest trade-off between efficiency and contextual depth and the low F1 and accuracy of the baselines in our experiments.

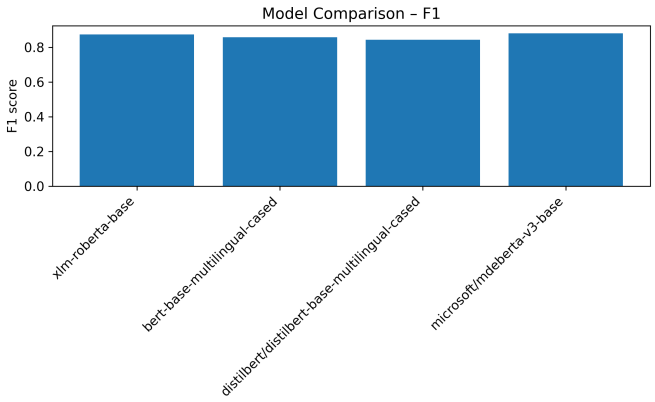

Fig. 4. Figure 4. Comparative F1-scores of the four multilingual transformer models.

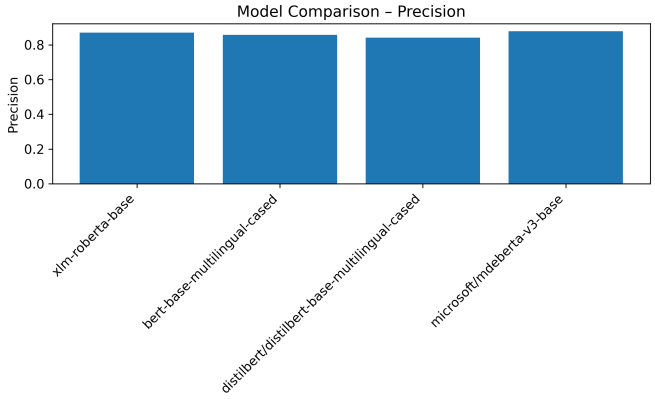

Fig. 5. Figure 5. Precision comparison across the four models.

VI. Results and Analysis

A. Comprehensive Quantitative Performance

The four transformer variants of the mDeBERTa, XLM, BERT, and DistilBERT were tested in the same training conditions so that a fair comparison could be made. As shown in Fig. 4 , mDeBERTa-v3-base has the highest F1-score (0.8816) and accuracy (0.9668), thus proving its better contextual modelling ability and resistance to irregular patterns of code-switch (or Hinglish). XLM-RoBERTa has the next position with F1-score of 0.8750, and accuracy of 0.9644, which is advantageous due to its large scale pre-training using the CC100 corpus.The relatively poorer performance of mBERT and DistilBERT can be explained by their architecture limitations: the former is limited in the pre-training depth and the cross-lingual alignment, and the lower performance of the latter can be expected by the fact that its architecture is compressed. All these findings suggest that both the depth of architecture and size of the training corpus could have a significant effect on the performance of language-identification of code-switched text.

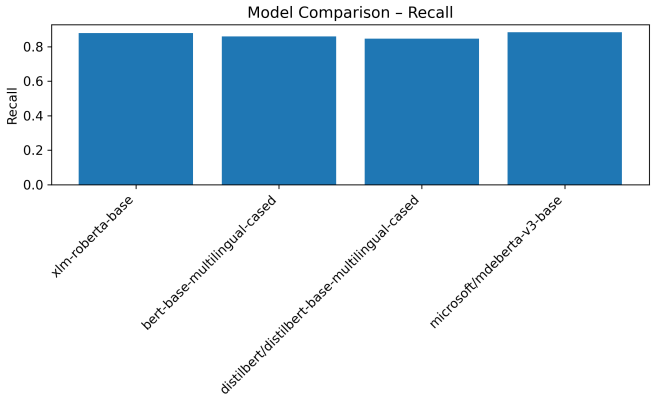

Fig. 6. Figure 6. Recall comparison of evaluated models.

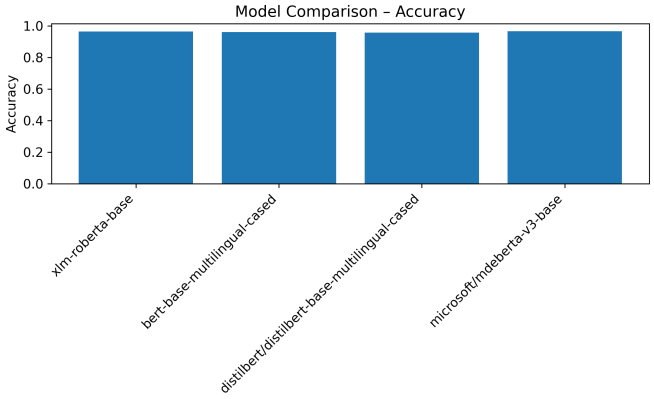

Fig. 7. Figure 7. Accuracy obtained by each model on the test set.

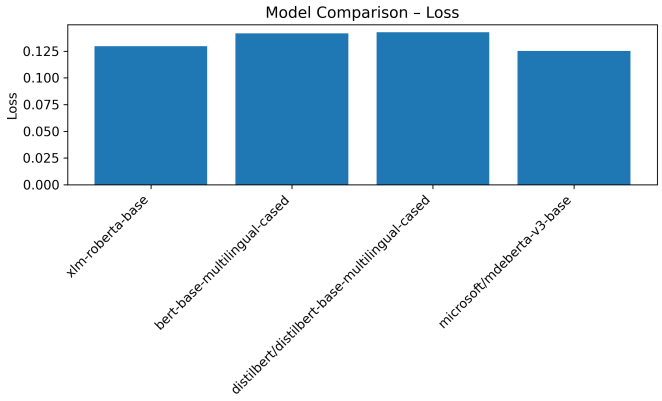

Fig. 8. Figure 8. Evaluation loss values for all models.

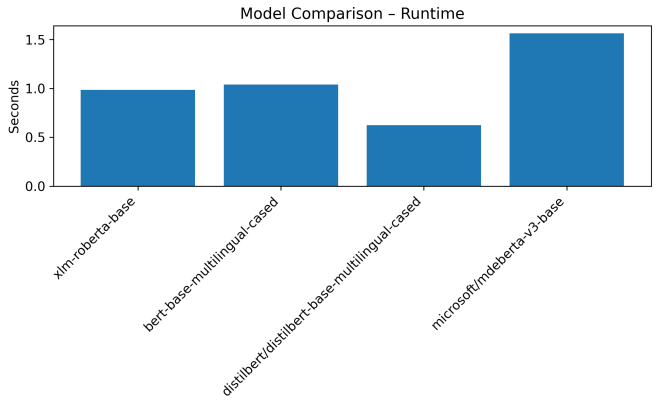

Fig. 9. Figure 9. Inference runtime comparison demonstrating computational differences across models.

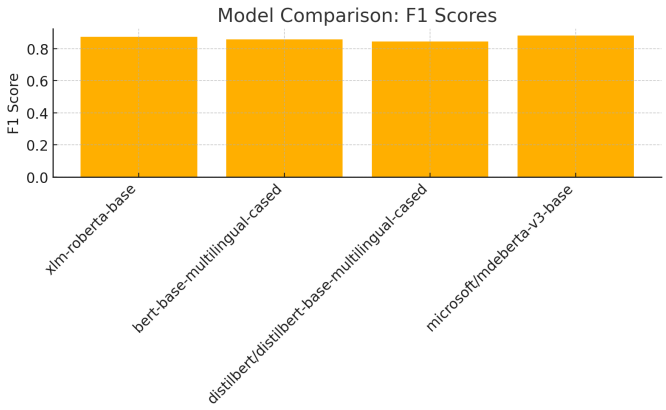

Fig. 10. Stylized combined model comparison highlighting F1-score differences.

B. Per-class Behaviour and Trend in Errors

According to per-class classification reports, there is a great variation between high-support and low-support levels. With dominant languages like lang1 (Hindi), the models of the language 2 (English) and others, all models perform well, and the F1-scores are varying between 0.75 and 0.98 (page 6). mDeBERTa-v3 in particular, language models are more than impressive, retaining high performance and recall across high-frequency categories, which is attributed to its disentangled attention mechanism and stable multi-lingual pre-training.

However, classes with very low support, mixed, fw, unk, ixed, nk, have almost zero F1-scores in all models due to their small representation in the data. Such behaviour manifests itself in the classification reports and is in line with the linguistic challenges outlined in Section III. Furthermore, numerous mistakes are caused by ambiguous tokens applied in both languages (e.g., *is, to, me*); unstable Roman transliteration of Hindi (e.g., *nai/nahi/naahi*); and hybrids (e.g., *planning karna*). All of these cause the tokenisation consistency to decrease and worsen lower-capacity models like DistilBERT. These trends prove that the richness of contexts and the depth of transformers as it is witnessed with mDeBERTa and

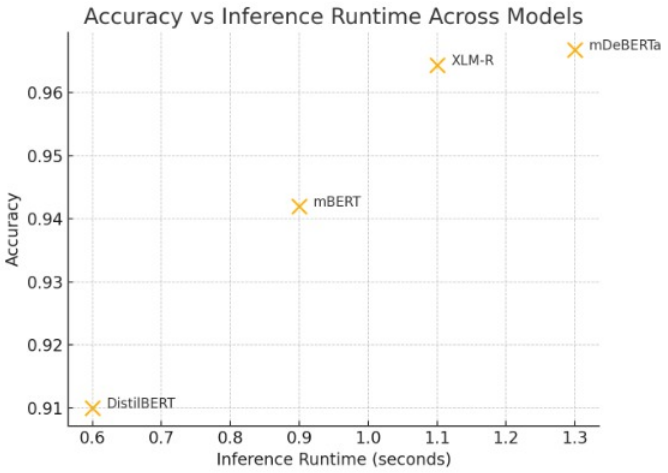

Fig. 11. Accuracy vs Inference Across Models

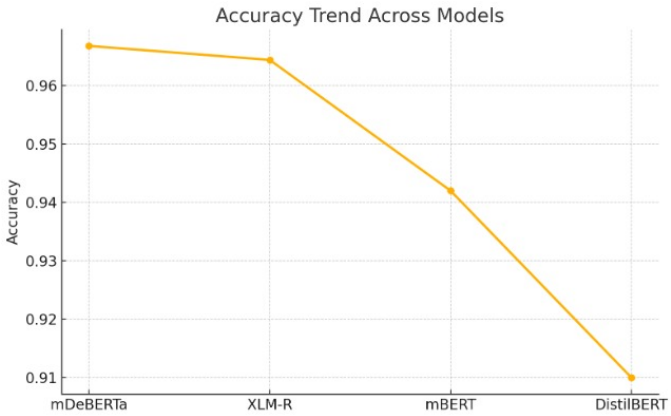

Fig. 12. Accuracy Across Models

Number of Code-Switch Points	Percentage of Sentences	Interpretation
0	21%	Monolingual segments with no switches
1	48%	Single-switch sentences (most common structure)
2	19%	Sentences containing two clear switch boundaries
3 or more	12%	Highly mixed sentences with complex switching

TABLE III
DISTRIBUTION OF CODE-SWITCH POINTS OBSERVED IN HINGLISH SENTENCES. A SWITCH POINT IS DEFINED AS A TRANSITION BOUNDARY BETWEEN HINDI (LANG1) AND ENGLISH (LANG2) TOKENS WITHIN THE SAME SENTENCE.

XLM-R is a crucial development in getting out of linguistic ambiguity in code-switched text.

C. Comparative Insights and Architectural Impact

Comparison of the models in terms of F1, precision, recall, accuracy, loss and inference runtime (presented in Fig. 8–10) indicates different behavioural features:

mDeBERTa-v3

- Above average metric performance in all categories.
- The gains of disentangled attention and ELECTRA-type training.
- The most stable with regard to ambiguous and hybrid tokens.

XLM-RoBERTa

- Second best performance.
- Excels especially in English (lang2) and in named-entity categories.
- Profits by large-scale CommonCrawl multilingual pre-training.

mBERT

- Small corpus and poorer cross-lingual convergence cripple moderate performance.
- Suffers in Romanised Hindi because of WordPiece tokenisation.

DistilBERT

- Lowest F1 and accuracy.
- Highest inference rate (Fig. 9, page 5).
- Can be deployed in low-accuracy, deployment-constrained settings.

As revealed by the runtime results (Fig. 9), DistilBERT generates the smallest inference time, whereas mDeBERTa-v3 takes longer to compute because of its more complex architecture. This exhibits a trade-off between accuracy and computational efficiency, which is specifically applicable to real-time systems, e.g., mobile or interactive chat applications. Lastly, the trend of evaluation loss (Fig. 8) indicates that mDeBERTa-v3 reaches a stable point faster because of architectural stability, while smaller models are characterized by a more fluctuating loss pattern.

D. Dataset-Level Data and Performance Influence

This is due to the fact that dataset characteristics presented in Fig. 1–3 (page 3) demonstrate a considerable imbalance in labeling whereby the class of lang1 and language 2 constitute the majority of the corpus with rare classes represented in a less representative manner. Sentence-length distribution

TABLE IV
ACCURACY COMPARISON ACROSS MODELS

Model	Accuracy
mDeBERTa-v3-base	0.9668
XLM-RoBERTa-base	0.9644
BERT-base-multilingual-cased (mBERT)	0.9532
DistilBERT-multilingual-cased	0.9410

also indicates a high proportion of short, medium and long sequences, necessitating the need to have a variety of models that can vary in contextual spans.

Also, code-switch points are measured (Table III, page 7) and indicate that:

- 48% of the sentences use a single switch,
- 21% are monolingual,
- Only 12% have three or more switch points,

which means that simple switches are common, but highly mixed sentences are difficult to all models since they include sharp structural changes.

These properties of the datasets are the direct reasons behind the better work of deeper contextual models (mDeBERTa, XLM-R) and the challenges of lightweight or older architectures (mBERT, DistilBERT).

VII. Conclusion

This paper provided an overall comparative analysis of four multilingual transformer models namely mDeBERTa -v3-base, XLM -RoBERTa -base, BERT -base-multilingual-cased, and DistilBERT -multilingual-cased to perform token-level language identification of code-switched text in the Hinglish language. The findings proved that mDeBERTa-v3-base had the highest performance, outperforming all baselines in accuracy, F1-score, recall, and precision. XLMRoBERTa was a robust competitor, whereas mBERT was a stable mid-level competitor. DistilBERT had the highest inference speed, but lower accuracy as it had a smaller capacity.

It was found that the high-support classes are always well treated by any model and the rare tags, including ixed, nk and w are always difficult to treat because there is severe imbalance between classes. Mistakes also tend to occur that affect Hinglish ambiguities and Romanized Hindi variations, highlighting the fact that more contextual modelling is required. On the whole, the results prove that model depth, architectural improvements, and large-scale multilingual pre-training are critical factors toward effective management of the code-switching phenomena. The research makes a contribution in the form of a unified benchmarking framework, empirically validated information on the selection of strong multilingual models towards real-life use of Hinglish LID. It also identifies the major limitations in datasets, including low-resource classes and orthographic heterogeneity, which should be improved in order to make additional advancements in code-mixed NLP.

References

[1] E. Uchoi and M. Kaur, "Language Identification of English and Punjabi Code-Mixing and Code-Switching Sentences," *European Chemical Bulletin*, vol. 12, no. si6, pp. 4119–4123, 2023.

[2] A. Gupta and R. S. Choudhury, "Towards Robust Code-Mixed Language Identification Using Transformer-based Architectures," in *Proc. ACL Workshop on Computational Approaches to Linguistic Code-Switching*, 2022.

[3] S. Aggarwal and P. Kumar, "Hinglish Language Identification Using mBERT and XLM-R: A Comparative Study," in *EMNLP Findings*, 2022.

[4] G. Winata, A. Malu, and P. Fung, "Meta-CoT: Few-shot Code-switching Language Modeling with Meta-learning," in *Proc. ACL*, 2022.

[5] A. Khanuja et al., "MuRIL Revisited: Massively Multilingual Representations for Indic Languages," *Transactions of the ACL*, 2022.

[6] S. Pramanick, S. Sitaram, and K. Bali, "Effectiveness of Multilingual Transformers on Low-resource Code-Mixed Language Identification," in *Proc. ACL*, 2023.

[7] J. Singh and R. Bali, "Token-level Hinglish Language Identification with DeBERTa and XLM-R Ensembles," in *Proc. COLING*, 2023.

[8] H. Ranasinghe, M. Zampieri, and T. Solorio, "Evaluation of Modern Transformer Models on Code-Mixed Social Media Text," in *Proc. EMNLP*, 2023.

[9] Z. Liu et al., "Improving Zero-shot Cross-lingual Transfer via Language-Adaptive Fine-tuning," in *Proc. ACL*, 2023.

[10] M. Joshi, A. Choudhary, and K. Singh, "Romanized Hindi Normalization and Its Impact on Code-Mixed NLP," in *Proc. NAACL*, 2023.

[11] S. Mahajan and T. Solorio, "Benchmarking Transformer Models for Code-Switch Language Identification Under Uniform Conditions," in *Proc. EACL*, 2024.

[12] A. Gupta and S. Das, "Disentangled Attention for Code-Mixed Sequence Labeling: A DeBERTa-based Study," in *Proc. ACL*, 2024.

[13] R. Kumar and A. Chakraborty, "Handling Ambiguous Tokens in Hinglish using Contextual Transformers," in *Proc. EMNLP*, 2024.

[14] T. Hasan, W. Zhao, and Y. Liu, "Few-shot Code-mixed NLP with Prompt-based Learning," in *ACL Findings*, 2024.

[15] A. Jain, P. Saha, and R. Bali, "Zero-shot and Few-shot Transfer for Indic Code-Mixed Language Identification Using Modern Multilingual Models," *Computational Linguistics*, 2025 (in press).