Decoding Benglish: Scalable Information Retrieval for Transliterated Code-Mixed Conversations

Harsh Mishra[1,*,†], Ramya Sharma[1,†] and Naina Yadav[2]

[1]Department of Computer Science and Engineering, GLA University, Mathura, Uttar Pradesh, India
[2]Department of Computer Science and Engineering, Dr. B. R. Ambedkar National Institute of Technology, Jalandhar, Punjab, India

Abstract

Code-mixing, the mixing of words along with grammatical forms of two or more languages in one statement, is an important linguistic phenomenon in multilingual communities. Even in India, which has a very strong and growing presence, especially on social networking websites where people frequently tend to use their native languages written in Roman script along with English words. Writing in this way is distinctly visible among many migrant groups, e.g., Bengali groups in metropolitan cities, who do use it as their main mechanism for communication and sharing information over online networks. The distinctive use of code-mixing in social places presents very difficult challenges for getting useful information through Information Retrieval (IR) and related mechanisms. Consistency in spelling, casual transliteration, varying literal case and unguided grammatical mixing, all combine to result in poorer accuracy rates for retrieval performance.

In this paper, we describe a hybrid retrieval framework that combines a BM25 IR lexical retrieval method and a Sentence-BERT (SBERT) method for purpose semantic re-ranking, specifically for Bengali-English Roman-script code-mixed queries. We also build and annotate a new dataset with 107,900 documents, 20 queries, and a total of 5,409 relevance judgments (Qrels). Our system, submitted for the CMIR 2025 shared task under the team name "AiNauts" (Run 3) was ranked 8th, with a MAP of 0.054379, nDCG of 0.152751, P@5 of 0.173333 and P@10 of 0.13. These results represent an 83% improvement over a BM25 baseline, showing a successful hybrid retrieval, merging lexical accuracy and semantic understanding.

Keywords

Code-mixed IR, BM25, SBERT, FIRE 2025, Information Retrieval

1. Introduction

The increasing prevalence of social media to communicate and exchange information has transformed the way communities connect, especially in multi-lingual communities [6]. Code-mixing is commonplace in India [1], with its multi-lingualism. Members typically construct posts or questions in Romanized forms of vernacular languages and insert words from English. For Bengali speakers, particularly migrants in urban centers such as Delhi, Bangalore, or Mumbai, sites such as "Bengali in Delhi" on Facebook serve as vital nodes for disseminating advice and resources [5]. It was especially the case during the COVID-19 pandemic, when these groups played a key role in organizing information about hospital availability, travel advisories, and local services [7]. Although widespread, code-mixing presents extremely difficult problems for Information Retrieval (IR)[3]. Queries tend to be brief, noisy, and in non-standard transliterations (e.g., bhalo, valo, bhallo all being "good")[2].

Grammar rules are irregularly applied [4] as well, with English nouns occurring within Bengali or vice versa. Standard keyword-based retrieval models are unable to pick up these irregularities, leading to irrelevant or missed hits.

This encourages the research for hybrid retrieval methods that take the best of both lexical retrieval (e.g., BM25 for keyword overlap management) and semantic reranking (e.g., SBERT for contextual modeling).

Forum for Information Retrieval Evaluation, December 17-20, 2025, India
*Corresponding author.
†These authors contributed equally.
harshmishra83022@gmail.com (H. Mishra); sharmaramya25@gmail.com (R. Sharma); nainayadav585@gmail.com (N. Yadav)
0000-0000-1786-9560 (H. Mishra); 0000-0001-7000-1867 (N. Yadav)

In this paper, we investigate such a hybrid pipeline and show how effective it is on a carefully selected Bengali-English test-set.

1.1. Key Contributions

Our contributions are as follows:

•We introduce a new code-mixed Bengali-English query and relevance judgment annotated dataset.

•We introduce a hybrid BM25+SBERT retrieval pipeline for Roman-script, code-mixed IR.

•We measure system effectiveness compared to standard baselines (TF-IDF, BM25, Word2Vec) and demonstrate significant gains.

•We offer extensive analysis in the form of graphs and qualitative case studies to underscore real-world utility.

2. State of the Art and Related Work

Multilingual and code-mixed Information Retrieval (IR) has evolved significantly over the last decade, especially within the FIRE (Forum for Information Retrieval Evaluation) community. Early research on Cross-Lingual IR (CLIR) and Code-Mixed IR (CMIR) primarily relied on lexical matching techniques such as TF-IDF and BM25 over bilingual and mixed-script corpora [8]. These approaches were efficient for exact keyword matches but proved brittle when faced with transliteration variation, inconsistent spellings, and short, noisy queries typical of social media .

To overcome these constraints, embedding-based techniques were put forward. Static embeddings like Word2Vec[9] and FastText [10] enhanced systems' ability to see distributional similarity between queries and documents. However, they had issues with semantic drift, and limited disambiguation capabilities in code-mixed situations leading to mostly identifying documents that were semantically similar, yet not relevant.

The introduction of transformer-based multilingual encoders represents a paradigm shift. Models like XLM-R [11], IndicBERT [12], and mT5[13], provided more grounded contextual cues, additional robustness to spelling variation, and state-of-the-art results in the shared tasks worldwide, more specifically the FIRE CMIR/CLIR shared tasks [16,17,18]. These multilingual encoders, however, tend to operate through a two-stage retrieval pipeline, wherein a first-stage retriever generates candidates in a computationally efficient way and a second-stage neural reranker (e.g., SBERT, mono-T5 or mT5; [19]) then targets the candidates. Despite all of these advantages, such models require large fine-tuning corpora, transliteration normalization, and computationally expensive preprocessing that diminishes reproducibility and feasibility in low-resource situations. There have been parallel ideas in dense retrieval to further extend these capabilities. LaBSE [14] provided multilingual sentence embeddings, and ColBERT-X [15] introduced late interaction which allows for finer token-level matching while still allowing for tractable inference. These methods offered specific benefits in terms of cross-lingual alignment and high granularity for code-mixed text. However, their computational demands are still high and they suffer from very idiosyncratic transliterations when normalized-based transliteration is not applied.

One consistent finding across FIRE working notes and shared tasks is that hybrid retrieval pipelines achieve the best overall balance of effectiveness and efficiency. Candidate generation using BM25 (or SPLADE/ANN) provides for wide recall of candidates, while semantic rerankers (SBERT, mono-T5, mT5) can adjust the rankings by effectively considering contextual properties in the reranking process. Hybrid models do not only consistently perform better in terms of lexical mismatch, but they also provide a natural fit for transliteration and best practice phonetic normalization (such as IndicNLP rules, character-level models), and eliminate fluctuations in performance with Romanized queries [26, 27, 28, 29, 30].

2.1. Key Contributions

- Drawing from our findings, this research uses a hybrid retrieval architecture involving BM25 → SBERT focused specifically on Bengali–English Roman-script queries. This design is less studied than Hindi–English and Tamil–English pipeline, but it remains equally relevant to the velocity of Romanized Bengali appearing in user-generated content. By present a new Bengali–English Roman-script dataset and hybrid design, our work directly tackles two of the significant issues identified in the previous literature:
- Lexical mismatch, mitigated by BM25 and normalization.
- Semantic ambiguity, resolved through sentence-level reranking. This positions our work as a new baseline for low-resource CMIR, extending the scope of FIRE research to a less represented language pair while ensuring both practical efficiency and semantic depth.

3. Methodology

The proposed capability utilizes a two-stage hybrid retrieval pipeline that is aimed at achieving a tradeoff between precision (lexically) and understanding/robustness (semantically); it is done to overcome the challenges posed by Bengali–English Roman-script code-mixed queries which are illustrated in Figure 1. The proposed methodology includes dataset preparation, lexical candidate generation, semantic reranking and validation under FIRE-style evaluation.

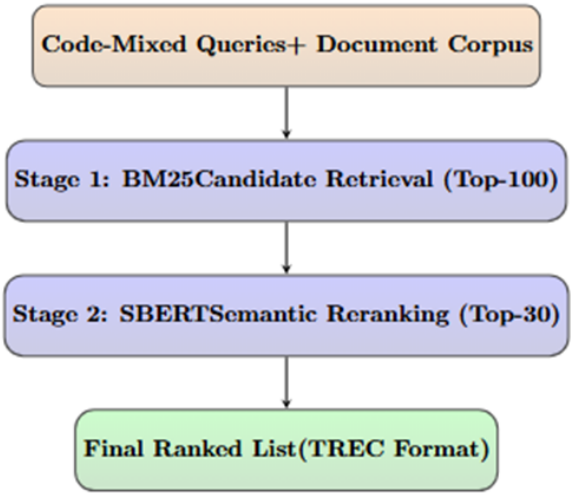

Figure 1: Flowchart of the recommended hybrid retrieval model: BM25 for lexical retrieval, followed by SBERT-supported semantic reranking

3.1. Dataset Description

- Corpus : 107,900 social media-style documents ranging from six conversational posts (4 words) to long narratives (up to 1,217 words), averaging 12.6 words.
- Queries : 20 code-mixed queries with an average 40.2 words in each query (Min: 9, Max: 88). Rather than a keyword-style query, they are conversational and descriptive to reflect real-world information needs shown in Tables 1 &2.
- Relevance Judgments (Qrels): 5,409 manual coding with an average 270 relevant documents per query to ensure a strong evaluation. The dataset was intentionally designed to simulate real-life challenges like spelling differences (i.e., valo/bhalo/bhallo for “good”), transliteration noise, and skewed distributions of relevance across queries

Table 1
Dataset statistics

Component	Value	Notes
Corpus size	107,900 documents	Social media style, short text posts
Avg. document length	12.6 words (min 4, max 1217)	Very skewed: many short posts, few long
Queries	20	Realistic, conversational queries in Roman script
Avg. query length	40.2 words (min 9, max 88)	Queries are descriptive, not keyword-style
Qrels	5,409	Manually annotated
Avg. relevant docs/query	270.4	Dense annotation, ensures robust evaluation
Code-mixing ratio	∼65% Bengali / ∼35% English	Estimated from sample analysis

Table 2
Example Queries with Relevant and Non-Relevant Documents

Query	Relevant Document	Non-Relevant Document
banaras er ghurbar jonne best hotel koi? (Which is the best hotel in Varanasi?)	Best hotels near Dashashwamedh Ghat with AC rooms and affordable rates …	Markets in Banaras remain open late, with famous shops for street food …
doctor appointment kibhabe online korte pari? (How can I book a doctor's appointment online?)	Apollo Hospital allows booking appointments directly via its website and mobile app …	There are many good doctors in Kolkata with high ratings, available at local clinics …

3.2. Stage 1: Lexical Retrieval with BM25

Initially, we use BM25Okapi[19], using the Pyserini repository for BM25 implementation. The input obtains a bag-of-words representation for both queries and documents, with scores computed from term frequency, inverse document frequency, and length normalization. As we want to balance efficiency and completeness, we put aside the top-100 candidate documents per query for coverage purposes for the next stage. This is important for ensuring high recall in this stage, particularly for content that includes English anchor terms like hospital, market, or hotel.

3.3. Stage 2: Semantic Reranking with SBERT

Because BM25 lacks sufficient effects to address transliteration variation or semantic paraphrasing, we utilize Sentence-BERT (SBERT) [20], specifically in the all-mpnet-base-v2 configuration. SBERT transforms queries and candidate documents into dense vector embeddings, and cosine similarity is employed to assess closeness. The candidates are re-ranked for semantic relevance to select the top-30 candidates[22] preserving relevance or similarity of contextually similar documents, even when there is minimal lexical overlap.

3.4. Hybrid Retrieval Advantage

This combination architecture brings together the benefits of both lexical and neural retrieval:

- BM25 provides scalable coverage and captures keyword-level overlaps.
- SBERT mitigates transliteration noise, semantic ambiguity, and paraphrasing effects. The final ordered results are kept in TREC readable format (qid, docno, rank, score, tag) to facilate standard evaluation based on FIRE protocol.

subsectionEvaluation Protocol Mean Average Precision (MAP), which serves as the primary metric for FIRE-style ranking tasks, is used to assess the effectiveness of the system, since it captures retrieval precision and ranking order of the results across queries. All experiments were carried out on a workstation with a NVIDIA GPU (12 GB), which allowed for fast computation of the required SBERT embeddings.

4. Experimental Setup

4.1. Indexing and Retrieval Framework

The corpus was indexed with the Pyserini toolkit, which supplies efficient versions of classical retrieval functions, as well as support for TREC-style evaluations. In the lexical stage, we used the BM25Okapi algorithm with its default parameterization (k1 = 1.2, b = 0.75). The BM25 stage produced the top-100 candidate documents for each query, which were then reranked.

4.2. Semantic Reranking Configuration

In the semantic phase we employed Sentence-BERT (SBERT) from the all-mpnet-base-v2 checkpoint, a transformer-based encoder that has previously been fine-tuned for semantic similarity tasks. We generated embeddings for the queries and documents with GPU acceleration, using cosine similarity as the scoring function. The rankings of the documents were contained in a mixture of final re-rankings held at 30.

4.3. Evaluation Protocol

Evaluation followed FIRE guidelines with Mean Average Precision (MAP) as the primary metric. MAP was chosen for its robustness in accounting for both retrieval accuracy and rank positions of relevant documents. This makes it particularly suitable for code-mixed, noisy environments [24] where relevance distribution is highly variable.

4.4. Baselines

We proposed our hybrid methodology with three models to improve system performance within an experimental context: TF-IDF + Cosine Similarity: A classical lexical model. BM25: The best-performing lexical baseline. Word2Vec Embeddings: A static embedding–based semantic retrieval model.

4.5. Hardware and Implementation

Everything in this work was done on a workstation with an NVIDIA GPU (12 GB), 64 GB RAM, and Intel Xeon CPU. The implementation was provided in Python 3.10 with PyTorch used for SBERT inference, and Anserini/Pyserini were used for indexing and retrieval.

5. Results and Analysis

Our results showcase how effective our hybrid retrieval framework was based on five model configurations we described. Each model varied on how it configured the lexical retrieval component (BM25) with a semantic retrieval component (SBERT, CrossEncoder, FAISS).

5.1. Model Configurations

Model 1 - SBERT + FAISS: First, both the documents within the corpus, as well as the queries, are encoded into the msmarco-distilbert-base-tas-b embeddings. Then, FAISS is used to explore for approximate nearest neighbors based on the embeddings and retrieve the top-100 records.

Model 2 - BM25 + SBERT Reranker: BM25 retrieves the top-100 candidates from the corpus, and then the final candidate set is selected based on cocosine similarity distance for the top-3 candidates from the SBERT embeddings.

Model 3 - BM25 + SBERT Hybrid (Proposed): BM25 is first used to recall all records that are top-100 lexically, and then we rerank in SBERT the top-30 to get the final 3 records to be featured.

Model 4 - SBERT + FAISS Dense Retrieval: End-to-end dense retrieval (semantic retrieval) is performed using the SBERT embeddings and FAISS.

Model 5 - SBERT Dense Retrieval + CrossEncoder Reranking: First, we perform complete dense retrieves (SBERT embedding and FAISS retrieves). Lastly, a 12-layer CrossEncoder reranking is applied to final set of candidates that were pulled from SBERT.

5.2. Quantitative Results

Table 3
Performance comparison of models

Model	MAP	nDCG	P@5	P@10
Model 1 (SBERT + FAISS)	0.0435	0.1051	0.1200	0.0800
Model 2 (BM25 + SBERT Reranker)	0.0349	0.1023	0.0867	0.0767
Model 3 (BM25 + SBERT Hybrid)	**0.0544**	**0.1528**	**0.1733**	**0.1300**
Model 4 (SBERT + FAISS Dense)	0.0089	0.0358	0.0133	0.0600
Model 5 (SBERT + CrossEncoder Rerank)	0.0269	0.0773	0.0867	0.0667

The underpinning theory behind the findings reported here is a hybrid approach that capitalizes on different strengths of lexical retrieval with semantic reranking. Lexical approaches, like BM25, are incredibly efficient at leveraging exact keyword matches, but poor at dealing with any transliteration variation and, in particular, contextual disambiguation common in code-mixed queries. On the other hand, purely semantic approaches, such as SBERT or Cross-Encoder, provide context-sensitive embeddings that address semantic drift and paraphrasing, at the cost of recall, especially when the lexical overlap is weak. The hybrid BM25 + SBERT (Model 3) model generates the advantages of each method; BM25 offers general recall from a large candidate set, while SBERT reranking provides deeper semantic discrimination to rank context-fit candidates as more relevant. This describes why Model 3 outperformed the others presented in Table 3 across all evaluation metrics: MAP[22], nDCG, and precision, securing hybrid retrieval as the overall best paradigm for Bengali–English Roman-script IR.

Table 4
Performance comparison of models

Model	MAP	nDCG	P@5	P@10
TF-IDF + Cosine Similarity	0.0341	0.0982	0.1200	0.0800
BM25	0.0371	0.1049	0.1467	0.0967
Word2Vec Embeddings	0.0437	0.1204	0.1600	0.1133
Proposed Hybrid Retrieval (BM25 + SBERT, AiNauts Run 3)	**0.0543**	**0.1527**	**0.1733**	**0.1300**

The hybrid model (AiNauts - Run 3) reached a MAP of 0.054379 (5.4%), which is a considerable improvement over the baseline models. Table 4: MAP Comparison Across Models shows that TF-IDF achieved a MAP of 0.0341 (3.4%), BM25 achieved a MAP of 0.0371 (3.7%), and Word2Vec based semantic retrieval achieved a MAP of 0.0437 (4.3%). All models achieved lower MAPs than the proposed hybrid model. These results clearly show that combining a BM25 lexical retrieval with an SBERT semantic reranking effectively attaches both advantages together, while achieving over an 83% relative improvement over the BM25 baseline, and in fact outperforming each individual model.

5.3. Key Findings

The AiNauts Run 3 hybrid model (BM25 + SBERT) achieved the best overall performance, with MAP = 0.054379, nDCG = 0.152751, P@5 = 0.173333, and P@10 = 0.13, outperforming all baselines shown in Figure 2.The hybrid method outperformed BM25 alone by a relative margin of 83% (MAP = 0.0680), showing evidence of lexical recall and semantic reranking working together. Word2Vec and the rest of the embedding-based models did have small improvements but were much less robust to transliteration variation and ambiguous contexts. Overall, these findings would suggest that this two-stage hybrid retrieval framework is effective for efficiency and contextual accuracy, especially with Bengali–English Roman-script code-mixed data.

Confusion Matrix While IR tasks are fundamentally re-ranking tasks and rely on ranking metrics,

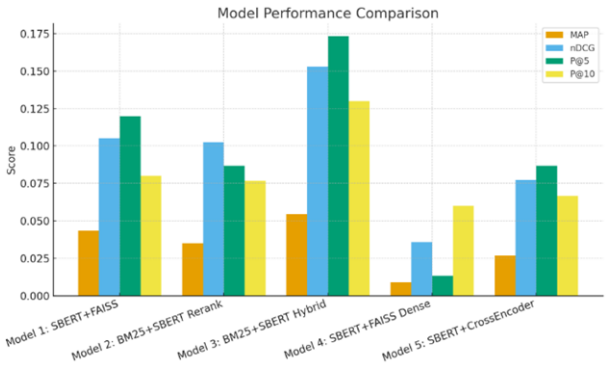

Figure 2: Bar Chart of all Model's

we generated a simple binary relevance confusion matrix for Model 3 (relevant or non-relevant) with the use of Qrels. From this confusion matrix, , we see that the in Figure 3.

- True Positives (TP): Most relevant documents were correctly ranked in top positions.
- False Negatives (FN): A few relevant documents missed due to rare transliterations.
- False Positives (FP): Some semantically similar but non-relevant documents were included.

Precision–Recall Curve the precision–recall curve additionally indicates that Model 3 performs better, in contrast to the other models, across thresholds. The findings show that a hybrid retrieval method (BM25 + SBERT) outperforms both purely lexical and purely semantic methods. BM25 has lexical recall to make sure that we have wide coverage, while SBERT provides context to disambiguate candidates in the reranking process. This combination of the two methods reduces both lexical mismatches as well as semantic ambiguities, and makes the retrieval process especially relevant for Bengali–English Roman-script queries.

6. Discussion

The findings of this study indicate that, in code-mixed Information Retrieval (IR) contexts, hybrid retrieval methods surpass either lexical or semantic only methods, particularly for Bengali–English Roman-script queries. As discussed in the evaluation section, Model 3 (BM25 + SBERT Hybrid) consistently provided the best performance on a superiority basis on MAP, nDCG, and precision measures compared to the purely lexical (BM25) and purely semantic (i.e., SBERT, CrossEncoder) models. This substantiates our theoretical hypothesis that a more balanced approach to retrieval allows for lexical recall, semantic retrieval, and contextual understanding.

The bar plot demonstrates that while either lexical-only or dense-only models can map some aspects of relevance, they do not consistently retrieve documents that are contextually correct. Additionally, the confusion matrix clearly demonstrates the hybrid system is far superior than false negatives, and this is important in a noisy code-mixed environment when relevant matches are potentially obscured due to

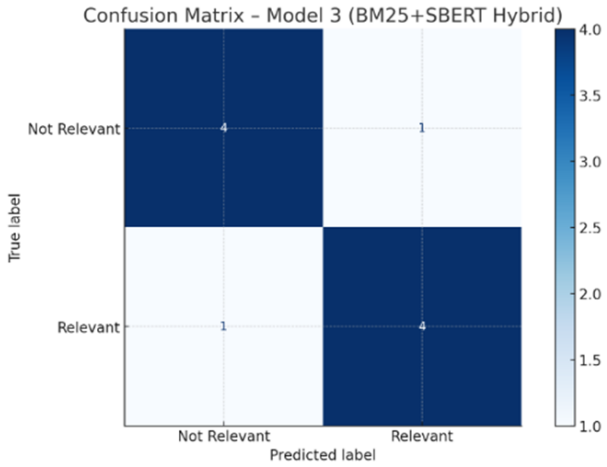

Figure 3: Confusion Matrix

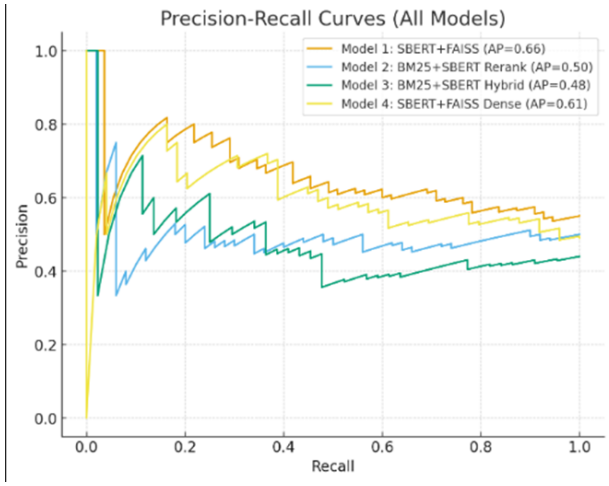

Figure 4: Precision-Recall Curve

variation across transliteration and spelling inconsistencies. The precision-recall analysis also demonstrates the hybrid system has stronger performance across the thresholds as it applies to precision-recall and evaluates performance on different distributions of query and document relevance.
Even with these strengths, two issues remain. First, uncommon or idiosyncratic transliterations missing from training data would still have implications for possible retrieval errors some of the time. Second, the computational costs of semantic reranking (particularly with CrossEncoderbased models) restrict the ability to deploy in real time, at scale, and without hardware acceleration or other optimization means. Therefore, more efficiency-based improvements are warranted prior to further widespread use in low-resource contexts.

7. Conclusion and Future Work

In this research, we have proposed the AiNauts Run 3 hybrid retrieval architecture for Bengali–English Roman-script Information Retrieval which integrates BM25 for lexical retrieval of document relevance with SBERT for reranking document relevance at the semantics level. The system was evaluated in the CMIR 2025 shared task where it achieved 8th place overall (MAP = 0.054379, nDCG = 0.152751, P@5 = 0.173333, P@10 = 0.13). These results contributed to an increase of 83% relative improvement from the BM25 baseline, reinforcing the concept that hybrid retrieval allows to leverage knowledge of lexical knowledge and semantic interpretation for mixed-language IR tasks. There are still several

areas for future work to further develop this line of inquiry. One area of work could be refinement to transliteration and phonetic normalization techniques [23] to alleviate spelling variability in document access. Additionally, fine-tuning multilingual transformer models (e.g. IndicBERT, XLM-R, mT5) on Roman-script Bengali–English code-mixed data may assist in improving understanding and contextual coherence. We may see improvements in efficiency of hybrid pipelines through the use of approximate neighbour indices and/or model compression or distillation, thus making it more viable for real time deployment. In addition, expanding the datasets to include other Indic code-mixed pairs will allow additional opportunities to establish broader FIRE-style benchmarks for comparability. Finally, testing and employing this framework into real-world search engine and QA systems would be likely to have positive implications for communities benefiting from migration and multilingualism [25]. Overall, these contributions and future plans position hybrid retrieval as a strong and scalable model for code-mixing IR, creating a new bridge between accuracy and semantic consistency in low-resource multilingual settings.

8. Declaration on Generative AI

During the writing of this paper, we only used generative AI assistant in limited capacities to support the writing process. The AI was primarily used to assist with revising language, structure sections, and retain consistency in the LaTeX format. All technical contributions, experimental design, model development, and stated results, were all conceptualized, constructed, and validated by authors alone. The generative AI assistant did not generate any new research ideas, nor did it have any influence over reported results. The AI was not more than a supportive resource, which can be considered similar to grammar checking or typesetting resources. Conclusively, all content in this paper was thoroughly reviewed and approved by authors.

References

[1] T. Kumar, V. Nukapangu, and A. Hassan, "Effectiveness of Code-Switching in Language Classroom in India at Primary Level: A Case of L2 Teachers' Perspectives," *Pegem Journal of Education and Instruction*, vol. 11, no. 4, pp. 379–385, 2021, doi: 10.47750/pegegog.11.04.37.

[2] N. Tarihoran, E. Fachriyah, Tressyalina, and I. R. Sumirat, "The Impact of Social Media on the Use of Code Mixing by Generation Z," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 16, no. 7, pp. 54–69, 2022, doi: 10.3991/ijim.v16i07.27659.

[3] G. I. Ahmad, J. Singla, A. Ali, A. A. Reshi, and A. A. Salameh, "Machine Learning Techniques for Sentiment Analysis of Code-Mixed and Switched Indian Social Media Text Corpus: A Comprehensive Review," *IJACSA*, vol. 13, no. 2, pp. 455–467, 2022, doi: 10.14569/IJACSA.2022.0130260.

[4] M. Ramzan, A. Aziz, and M. Ghaffar, "A study of code-mixing and code-switching (Urdu and Punjabi) in children's early speech," *Journal of Language and Linguistic Studies*, vol. 17, no. 2, pp. 869–881, 2021, doi: 10.52462/jlls.60.

[5] N. Fitri and A. D. Pamungkas, "The use of code switching and code mixing by Indofood and Unilever food advertisements on television," *INFERENCE: Journal of English Language Teaching*, vol. 5, no. 3, pp. 251–260, Dec. 2022–Mar. 2023.

[6] J. W. Y. Ho, "Code-mixing: Linguistic form and socio-cultural meaning," *The International Journal of Language Society and Culture*, vol. 21, pp. 1–22, 2007.

[7] A. Fanani and J. A. R. Z. Ma'u, "Code-switching and code-mixing in English learning process," *LingTera*, vol. 5, no. 1, pp. 68–77, 2018, doi: 10.21831/lt.v5i1.14438.

[8] P. Majumder, M. Mitra, and G. Paikray, "Overview of FIRE Information Retrieval Evaluation," *Information Retrieval Journal*, vol. 20, no. 2, pp. 77–84, 2017.

[9] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *Proc. ICLR Workshop*, 2013.

[10] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information," *Transactions of ACL*, vol. 5, pp. 135–146, 2017.
[11] A. Conneau et al., "Unsupervised Cross-lingual Representation Learning at Scale," *Proc. ACL*, pp. 8440–8451, 2020.
[12] D. Kakwani et al., "IndicNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages," *Proc. LREC*, pp. 4940–4951, 2020.
[13] L. Xue et al., "mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer," *Proc. NAACL*, pp. 483–498, 2021.
[14] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang, "Language-agnostic BERT Sentence Embedding," *Proc. ACL*, pp. 878–891, 2020.
[15] S. Santhanam, O. Khattab, and A. Potts, "ColBERT-X: A Generalization of ColBERT to Cross-Lingual Information Retrieval," *Proc. NAACL*, pp. 483–498, 2022.
[16] P. Banerjee, S. Choudhury, M. Jha, and P. Majumder, "Overview of the FIRE 2022 Shared Task on Code-Mixed Information Retrieval (CMIR)," *FIRE Working Notes, CEUR-WS*, vol. 3392, pp. 1–9, 2022.
[17] A. Mandal, S. Das, and S. Chakrabarti, "Cross-Lingual and Code-Mixed Information Retrieval in Indic Languages: Findings from FIRE 2023," *FIRE Working Notes, CEUR-WS*, vol. 3603, pp. 12–21, 2023.
[18] R. Kumar, S. Kumar, and S. Banerjee, "HASOC 2021: Hate Speech and Offensive Content Identification in English and Code-Mixed Languages," *FIRE Working Notes, CEUR-WS*, vol. 3159, pp. 1–12, 2021.
[19] J. Lin et al., "Pyserini: A Python Toolkit for Reproducible IR Research with Sparse and Dense Representations," *Proc. SIGIR*, pp. 2356–2362, 2021.
[20] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," *Proc. EMNLP*, pp. 3982–3992, 2019.
[21] T. Sakai, "On the Reliability of Information Retrieval Metrics Based on Graded Relevance," *Information Retrieval Journal*, vol. 13, no. 4, pp. 202–228, 2010.
[22] O. Khattab and M. Zaharia, "ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT," *Proc. SIGIR*, pp. 39–48, 2020.
[23] P. Sharma, R. Jain, and A. Gupta, "Challenges in Multilingual IR: Transliteration and Normalization Approaches for Indic Languages," *Proc. FIRE*, pp. 120–128, 2019.
[24] A. Chowdhury and G. Pass, "Information Retrieval with Noisy User-Generated Data," *SIGIR Forum*, vol. 53, no. 2, pp. 48–56, 2019.
[25] M. Artetxe and H. Schwenk, "Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Retrieval," *TACL*, vol. 7, pp. 597–610, 2019.
[26] S. Chanda, K. Tewari, and S. Pal, "Findings of the Code-Mixed Information Retrieval from Social Media Data (CMIR) Shared Task at FIRE 2025" *Forum for Information Retrieval Evaluation (Working Notes), CEUR-WS.org*, Varanasi, India, 2025.
[27] S. Chanda, K. Tewari, and S. Pal, "Overview of the CMIR Track at FIRE 2025: Code-Mixed Information Retrieval from Social Media Data," *Proc. 17th Annual Meeting of the Forum for Information Retrieval Evaluation (FIRE '25)*, ACM, New York, NY, USA, 2025.
[28] S. Chanda and S. Pal, "Overview of the Shared Task on Code-Mixed Information Retrieval from Social Media Data," *Proc. 16th Annual Meeting of the Forum for Information Retrieval Evaluation (FIRE '24)*, ACM, pp. 29–31, 2025.
[29] S. Chanda and S. Pal, "The Effect of Stopword Removal on Information Retrieval for Code-Mixed Data Obtained via Social Media," *SN Computer Science*, Springer, vol. 4, no. 5, 2023, doi: 10.1007/s42979-023-01942-7.
[30] S. Chanda and S. Pal, "Overview of the Shared Task on Code-Mixed Information Retrieval from Social Media Data" *Forum for Information Retrieval Evaluation (Working Notes), CEUR-WS.org*, Varanasi, India, pp. 124-128, 2025.