Findings of the Code-Mixed Information Retrieval from Social Media Data (CMIR) Shared Task at FIRE 2025

Supriya Chanda[1,*], Krishna Tewari[2] and Sukomal Pal[2]

[1] *Bennett University, Greater Noida, Uttar Pradesh, India*
[2] *Indian Institute of Technology (BHU) Varanasi, Uttar Pradesh, India*

Abstract

The increasing use of multilingual and code-mixed communication on social media presents unique challenges for Information Retrieval (IR), especially in low-resource languages such as Bengali. To foster research in this direction, we organized the CMIR-2025 shared task, the second edition of the Code-Mixed Information Retrieval challenge. Building on the initial CMIR-2024 task, where only Roman-script Bengali was considered, this year's edition introduces a more realistic and complex setting by retaining Bengali words in their native script. The resulting dataset contains mixed-script Bengali–English text, requiring participating systems to retrieve relevant comments for a given query from social media discussions. Eight teams participated, submitting a total of 26 runs using lexical models, neural rankers, and fusion-based approaches. Evaluation using MAP, nDCG, P@5, and P@10 reveals that fusion and hybrid retrieval systems consistently outperform standalone models, indicating the importance of combining lexical and semantic signals for handling noisy code-mixed data. This paper presents the dataset, task design, evaluation results, and key insights that highlight open challenges and future research directions for mixed-script IR.

Keywords

Code-Mixed, Bengali, English, Information Retrieval, Social Media

1. Introduction

The growing volume of multilingual and code-mixed content on digital platforms continues to pose challenges for both Natural Language Processing (NLP) and Information Retrieval (IR), particularly in linguistically diverse regions such as India. Code-mixing, the practice of blending two or more languages within the same utterance, has become a natural mode of communication on social media, where users frequently switch between English and local languages. Bengali–English and Hindi–English are among the most prominent combinations, often written using non-native or mixed scripts. Traditional IR systems, designed primarily for monolingual and well-structured text, struggle in such environments due to spelling variation, informal syntax, transliteration noise, and lack of linguistic standardization.

In Bengali-speaking communities, the use of Roman script to write Bengali is extremely common in social media communication. This phenomenon, while convenient for users, introduces additional complexity for retrieval models, since transliteration is often inconsistent and ambiguous. Such text frequently contains colloquial forms, code-switching patterns, and morphologically rich Bengali tokens alongside English words. As a result, core IR components such as tokenization, vocabulary matching, ranking, and semantic similarity estimation become significantly more difficult.

Research in code-mixed NLP has progressed in recent years, with studies addressing language identification [1], sentiment analysis [2, 3, 4], hate speech detection [5], transliteration normalization, and sarcasm detection [6]. However, IR for code-mixed languages, especially low-resource ones like Bengali, remains relatively underexplored. Developing retrieval systems that can accurately rank relevant responses in noisy, multilingual discussions is still an open research problem.

To address this gap, we introduced the first shared task on CMIR in 2024, focusing on Bengali–English queries and comments primarily in Roman script. Although several teams registered for CMIR-2024,

Forum for Information Retrieval Evaluation, December 17-20, 2025, India

[*] Corresponding author.

suplife24@gmail.com (S. Chanda); krishnatewari.rs.cse24@itbhu.ac.in (K. Tewari); spal.cse@itbhu.ac.in (S. Pal)
0000-0002-6344-8772 (S. Chanda)

only two teams [7, 8] successfully submitted system runs, indicating that the task was both new and challenging. The initial edition demonstrated the feasibility of building IR systems for code-mixed text, yet also revealed limitations in current approaches, leaving ample scope for improvement.

In this paper, we present CMIR-2025, the second edition of the shared task. Building upon the previous setup, this edition increases complexity by retaining Bengali tokens in their native Bengali script instead of fully transliterating them into Roman form. Consequently, the dataset now contains script-level code-mixing, introducing challenges in token alignment, indexing, embedding representation, and cross-script retrieval. Participants are required to retrieve and rank relevant comments for a given query from a collection of code-mixed Bengali–English social media posts, with both training and test data provided.

This shared task aims to promote scalable, multilingual retrieval methodologies capable of handling real-world mixed-script communication. Through CMIR-2025, we provide the research community with a benchmark to explore retrieval strategies, fusion methods, cross-lingual representations, and script-aware ranking models. We hope this second edition drives further progress toward more inclusive and linguistically robust IR systems for social media environments.

This paper discusses the various models submitted to the shared task and the results of the participating teams. The rest of the article is orchestrated as follows: Section 2 describes the shared task. The related works are discussed in Section 3. Section 4 discusses about the dataset. Section 5 summarizes the systems and the methodologies used in each participating team for the shared task and highlights the features of each model. The analysis of the results and findings of the methodologies submitted by the participants are presented in Section 6. Concluding remarks are presented in Section 7.

2. Task Description

The objective of the task [1] is to automatically estimate the relevance of a document with respect to a user query in a code-mixed environment, primarily involving English, Bengali, and Romanized Bengali text. Each document must be classified as *relevant* or *non-relevant* to the given query, followed by ranking based on relevance scores. This setup introduces additional challenges such as multilingual mixing, transliteration variations, informal spelling patterns, and non-standard orthography. Therefore, an effective retrieval model must capture semantic similarity between the query and documents while being robust to cross-lingual noise and script variations.

In CMIR, the query and the document collection are not constrained to a single language-script pair (see Figure 1). Instead, both may exhibit mixing of multiple languages and scripts in any proportion. Let the query space be represented as

$$q \in \mathcal{Q} = \bigcup_{m \geq 2,\, n \geq 1} \left(\mathbb{L}^{(m)} \times \mathbb{S}^{(n)} \right),$$

where

$$\mathbb{L}^{(m)} = \{\ell_1, \ell_2, \ldots, \ell_m\}, \qquad \mathbb{S}^{(n)} = \{\sigma_1, \sigma_2, \ldots, \sigma_n\}$$

denote the set of m possible languages and n possible scripts respectively.
Similarly, the document collection is defined as

$$\mathcal{D} = \bigcup_{(\ell,\sigma) \in \mathbb{L}^{(m)} \times \mathbb{S}^{(n)}} \mathcal{D}_{\ell,\sigma},$$

where $\mathcal{D}_{\ell,\sigma}$ represents the subset of documents written in language ℓ using script σ. The retrieval task in CMIR aims to find the most relevant documents from $\mathcal{D}$ for a given query q, despite language and script heterogeneity.

[1] https://cmir-iitbhu.github.io/cmir/index.html

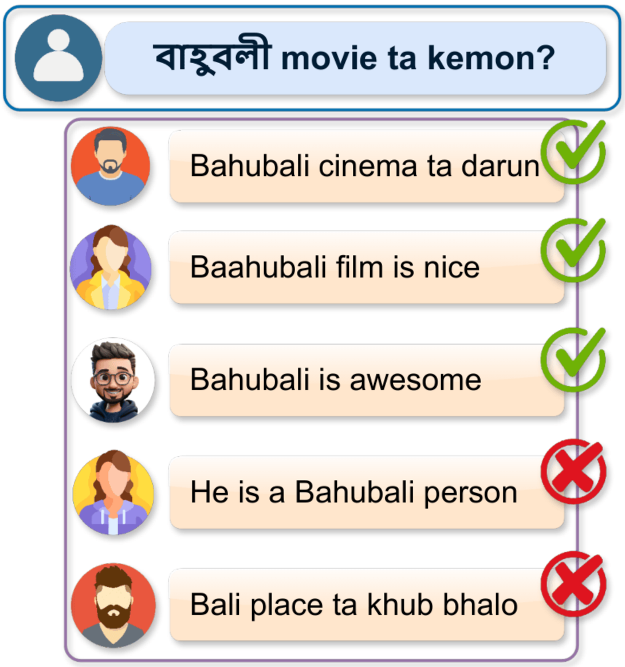

Figure 1: Illustration of the Code-Mixed Information Retrieval task. A query written in mixed English-Bengali is matched against a pool of user-generated comments/documents. Relevant items are expected to be identified and ranked above non-relevant items.

3. Related Work

Research in Information Retrieval (IR) has evolved remarkably over the past decades, beginning with early rule-based systems and progressing toward hybrid and neural retrieval pipelines. Foundational work in IR introduced vector space modeling and TF–IDF weighting [9], followed by probabilistic ranking frameworks including BM25 [10] and language modeling approaches [11], which continue to serve as high-performing sparse baselines in IR benchmarks [12]. These models, while effective for monolingual text, struggle in multilingual and noisy social media environments due to exact-matching dependence and vocabulary mismatch.

With the growth of multilingual content in India, early studies shifted toward mixed-script and Cross-Lingual IR (CLIR). Works such as [13, 14] explored retrieval across English and Indic languages using bilingual dictionaries, suffix-based stemming, and transliteration. Later FIRE tracks formalized IR evaluation for Indian scripts, including mixed-script Hindi queries [15, 16]. These early systems primarily relied on TF–IDF, BM25, Boolean retrieval, and stemming-based normalization, reporting low MAP values (≈0.15–0.18), emphasizing the difficulty of noisy user-generated text.

Subsequent work focused on resource creation and handling transliteration variations at scale. Studies on Hindi-English and Bengali corpora [16, 17] introduced datasets for CMIR, demonstrating improvements through stopword design and normalization. Research also examined token-labeling approaches using CRF and supervised classifiers for mixed queries [18, 19], with improvements under mixed-script indexing and fuzzy normalization. Several works evaluated query expansion, phonetic encoding, and Roman-to-native script mapping to reduce spelling inconsistency [20]. Despite improvements, performance remained limited due to sparse vocabulary coverage and informality of social text.

The shift toward dense retrieval began with the advent of deep contextual models such as BERT, RoBERTa and transformer-based ranking frameworks. Sentence-BERT, ColBERT, and ColBERTv2 improved ranking quality via semantic representation and late-interaction token matching. Multilingual encoders such as mBERT, XLM-R, IndicBERT/IndicBART and LaBSE enabled cross-lingual transfer, forming the foundation for multilingual IR systems. However, dense models required GPU-heavy training, domain alignment, and transliteration normalization to generalize across noisy scripts. Two-stage retrieval pipelines (BM25 → neural reranker) emerged as a practical balance between efficiency and semantic matching [21].

The FIRE shared tasks accelerated progress in CMIR through dataset releases [22] and track-based evaluation. Overviews presented in FIRE-2024 [23, 24] highlighted persistent challenges including code-switching boundaries, orthographic inconsistency, Romanization, slang, short queries, and lack of standard spelling. Further benchmarks like GLUECoS, LinCE, Cocktail and CoIR reported sensitivity in dense models under cross-script drift, motivating hybrid fusion approaches.

Normalization, stopword engineering, transliteration and phonetic indexing (e.g., Hindex) proved useful for Indic CMIR, showing 15–16% MAP improvements in retrieval [22]. IndicIRSuite further contributed multilingual IR resources, including IndicColBERT. Meanwhile, hybrid systems combining BM25, language models, SPLADE, and Reciprocal Rank Fusion emerged as reliable baselines, consistently outperforming single-model retrieval.

4. Dataset

In this work, we utilize the same dataset that was introduced in CMIR 2024 [23], with significant extension and refinement. The earlier version primarily consisted of code-mixed text where the Bengali content was transliterated into Roman script to ensure uniformity. However, this restricted the representation of native Bengali script usage in real-world social media communication. To enhance linguistic diversity and increase the complexity of retrieval, the current version retains Bengali words in their original Bengali script instead of transliteration. This modification introduces a more realistic multilingual environment and poses additional challenges for information retrieval systems dealing with script variation.

Similar to the previous data collection protocol, the dataset was curated from Facebook public groups and pages with active participation from Bengali-speaking communities. Queries correspond to the main posts in the discussion thread, while user comments associated with each post are treated as documents. The retrieval task involves identifying which comments are relevant to a given post and ranking them accordingly.

The dataset consists of a total of 50 queries paired with 107,900 documents. For experimentation, we provide 20 queries for training, along with corresponding QRels relevance annotations, and 30 queries for testing. Notably, training queries predominantly contain Romanized Bengali-English text, whereas the test set includes a mix of Bengali script and Roman script, making the evaluation more challenging and reflective of natural code-mixed behavior.

5. Methodology

In total, 23 teams registered for the CMIR-2025: Code-Mixed Information Retrieval shared task. And 8 teams were able to submit their system runs.

NITA_CMIR [25]: The preprocessing starts by cleaning up the messy, informal spellings found in both English and Romanized Bengali text. Common shortcuts like “gd → good" or “6ilo → chilo" are replaced using custom dictionaries, and the text is then tokenized and matched with its corrected forms. To catch additional spelling variations, a second pass with RapidFuzz maps noisy or unfamiliar words to their closest valid match. After this cleanup, the data is reformatted to work smoothly with PyTerrier for indexing and retrieval. Finally, several classic retrieval models: TF-IDF, BM25, PL2, InL2, and Hiemstra_LM, are run on each query. Their results are merged using Reciprocal Rank Fusion (RRF), which boosts documents that consistently appear near the top across multiple models, creating a stronger and more reliable ranking.

MUCS [26]: The study employs a combination of three classical IR models - BM25, Dirichlet Language Model, and Hiemstra Language Model whose ranked outputs are merged using Reciprocal Rank Fusion

(RRF). This strategy leverages the complementary strengths of each model to reduce query–document mismatch in noisy multilingual settings. A crucial aspect of the system design is an improved indexing configuration that disables stemming and stopword removal, ensuring that transliterated Bengali tokens and informal variations are preserved rather than discarded. Experiments conducted using PyTerrier show that while individual models perform well, the fused system consistently delivers superior effectiveness across evaluation metrics.

CodeWeavers [27]: This is a two-stage re-ranking framework. In the first stage, the first k documents were retrieved using a lexical model (through BM25). In the second stage, a zero-shot bi-encoder was used to model semantic similarity on the query and the candidate documents and re-rank the first k results from the lexical model. The framework remains lightweight while capturing lexical and semantic signal.

Defense_NLP [28]: This is a multi-stage retrieval and reranking architecture combining lexical (BM25), semantic (E5), and cross-encoder (MiniLM) models. After that it refined through a dynamic XGBoost-based meta-learner. This modular design allows the system to balance surface-level token overlap with deep semantic understanding while remaining lightweight and deployable in resource-constrained environments.

A key contribution of this work is a systematic examination of preprocessing techniques, model-level design choices, and score fusion strategies for code-mixed retrieval. Extensive ablations reveal that standard NLP techniques such as stemming, lemmatization, phonetic normalization, and named-entity abstraction often reduce performance in informal, transliterated settings. The author also show that larger multilingual models such as XLM-R and coupled training procedures like contrastive or joint fine-tuning consistently underperform due to overfitting and noise sensitivity. In contrast, the MiniLM-based reranker and learned fusion mechanism offer significantly higher robustness and top-k accuracy.

mixmatch IR [29]: The author propose a hybrid retrieval framework that integrates BM25 and a triplet-tuned Sentence Transformer model using Reciprocal Rank Fusion (RRF). the approach leverages the complementary strengths of sparse and dense retrieval, ensuring robust performance on noisy Banglish social media data.

NLPFusion [30]: This model employs an ensemble-based approach that integrates three ranking models, fused using CombSUM, CombMNZ, and Reciprocal Rank Fusion (RRF) to utilize complementary strengths and enhance retrieval robustness.

IRSolver [31]: The work and findings presented in this paper are purely experimental and a brute-force approach was adopted. The corpus, queries and qrels provided were checked for null values or empty fields and then converted into pandas dataframe for better readability. Two experiments were then conducted on these files, one where the queries were not expanded and retained in their original form, and the other where queries were expanded using Soundex, Phonix and Hindex phonetic algorithms for the individual language tags obtained from language identification, to get the best query expansion which gives the best performance in retrieval amongst all possible configurations. This was done to account for different spellings but same pronunciation of query words, a common yet complex feature of transliterated Indic languages. Throughout the literature survey, both traditional IR algorithms like BM25, TF-IDF, InL2 etc. and neural IR models like Learning-To-Rank models, Prompt-based methods, DeepCT etc were explored to understand their feasibility for code-mixed Indic language data. Eventually model selection was narrowed down to BM25 and ColBERT (a late-interaction based IR model). he corpus was indexed using the ColBERT indexer provided by PyTerrier platform, for it to be able to parse the contents properly. In both experiments, the queries were checked against the corpus using only BM25, ColBERT and a two-stage retrieval engine consisting of BM25 followed by ColBERT-based reranking were reported. The ColBERT model, however, is not trained on the training data, rather the pre-trained checkpoints were used. All experiments were carried out using the PyTerrier framework and the entire code was written in Python.

AiNauts [32]: The authors propose a two-stage hybrid retrieval framework that integrates lexical retrieval (BM25) with semantic reranking (SBERT). The system first retrieves the top-100 candidates using BM25, ensuring high lexical recall, and then applies SBERT (all-mpnet-base-v2) to rerank the top-30 using cosine similarity. Experiments compare this hybrid method against several baselines, including TF-IDF, Word2Vec, SBERT-FAISS dense retrieval, and Cross-Encoder reranking.

6. Results and Discussion

The performance of all participating teams was evaluated using four metrics: Mean Average Precision (MAP), nDCG, P@5, and P@10. However, the ranking was decided based on MAP. Teams were allowed to submit multiple runs with variations in model design and parameter settings. Table 1 presents the complete evaluation scores, where the best submission from each team is highlighted in grey. The leaderboard reflects diverse approaches, ranging from traditional lexical retrieval to neural ranking, fusion pipelines, and hybrid semantic models.

Team Name	Submission File	MAP Score	nDCG Score	P@5 Score	P@10 Score
AiNauts	Run 1	0.043466	0.105056	0.120000	0.080000
AiNauts	Run 2	0.034908	0.102270	0.086667	0.076667
AiNauts	Run 3	0.054379	0.152751	0.173333	0.130000
AiNauts	Run 4	0.008905	0.035781	0.013333	0.060000
AiNauts	Run 5	0.026915	0.077313	0.086667	0.066667
Defense_NLP	Run 1	0.178594	0.366472	0.373333	0.290000
Defense_NLP	Run 2	0.187175	0.376756	0.393333	0.276667
NITA_CMIR	Run 1	0.151773	0.415057	0.300000	0.236667
MUCS	Run 1	0.081828	0.292136	0.213333	0.156667
MUCS	Run 2	0.211792	0.485517	0.420000	0.300000
MUCS	Run 3	0.211792	0.485517	0.420000	0.300000
MixMatch IR	Run 1	0.123289	0.375620	0.293333	0.210000
MixMatch IR	Run 2	0.110594	0.318225	0.233333	0.166667
IRSolver	Run 1	0.063702	0.147071	0.226667	0.156667
IRSolver	Run 2	0.057110	0.139316	0.206667	0.156667
CodeWeavers	Run 1	0.156270	0.341038	0.280000	0.230000
CodeWeavers	Run 2	0.154625	0.276684	0.380000	0.283333
NLPFusion	Run 1	0.136089	0.319129	0.286667	0.233333
NLPFusion	Run 2	0.136089	0.319129	0.286667	0.233333
NLPFusion	Run 3	0.136089	0.319129	0.286667	0.233333
NLPFusion	Run 4	0.092505	0.246699	0.233333	0.170000
NLPFusion	Run 5	0.092505	0.246699	0.233333	0.170000

Table 1
Performance Scores of Team Submissions with Best Runs Highlighted

NITA_CMIR submitted a single run, achieving MAP = 0.1518, nDCG = 0.4151, P@5 = 0.30, and P@10 = 0.2367. Despite no multiple iterations, the system performed strongly, demonstrating the effectiveness of their chosen retrieval strategy for code-mixed IR.

MUCS emerged as the top-performing team, with their best run (Run 2/3 tie) obtaining MAP = 0.2118, nDCG = 0.4855, P@5 = 0.42, and P@10 = 0.30. Their results reveal that an effective rank-fusion of classical lexical retrieval models can outperform more complex neural architectures in code-mixed search. This highlights the continued relevance of probabilistic IR techniques in multilingual settings.

CodeWeavers ranked third overall, with their best submission scoring MAP = 0.1563, nDCG = 0.3410, P@5 = 0.28, and P@10 = 0.23. Their hybrid approach combining term-based retrieval and semantic understanding proved effective, demonstrating that zero-shot semantic reasoning can complement indexed retrieval for noisy code-mixed text.

Defense_NLP secured the second position, achieving MAP = 0.1872, nDCG = 0.3768, P@5 = 0.3933, and P@10 = 0.2767. Despite having only 20 labeled queries for training, the model generalized well, providing strong results in a low-resource setup. Their analysis emphasizes practical insights for multilingual IR, including design pitfalls and data preprocessing considerations.

MixMatch IR achieved MAP = 0.1233, nDCG = 0.3756, P@5 = 0.293, and P@10 = 0.21. The team showed that Reciprocal Rank Fusion (RRF) leads to noticeable gains over isolated models, making fusion-based approaches promising for robust code-mixed retrieval.

NLPFusion submitted five runs, with Run 1/2/3 yielding identical top performance: MAP = 0.1361, nDCG = 0.3191, P@5 = 0.287, and P@10 = 0.233. The stable results across multiple submissions indicate a well-designed ensemble ranking framework that effectively handles linguistic diversity in Bengali-English mixed search contexts.

IRSolver achieved a best score of MAP = 0.0637, nDCG = 0.1471, P@5 = 0.227, and P@10 = 0.157. Their experiments show that query expansion improves retrieval performance, while single-stage neural rankers require more sophisticated optimization. A two-stage retrieval pipeline produced the most reliable outcome.

AiNauts submitted five runs, with Run 3 performing the best: MAP = 0.0544, nDCG = 0.1528, P@5 = 0.1733, and P@10 = 0.13 marking an 83% improvement over the BM25 baseline. Their BM25 + SBERT hybrid approach validated that semantic similarity incorporation helps retrieve contextually relevant documents in noisy, code-mixed environments.

So the overall observations from the above table is that MUCS leads the leaderboard followed by Defense_NLP and CodeWeavers. Across submissions, fusion or hybrid models consistently outperformed standalone retrieval systems, highlighting the need for combining lexical and semantic signals in code-mixed IR. The results suggest that variability in script representation, informal spelling, and multilingual code-mixing continue to make CMIR a challenging problem space, opening avenues for further advancement.

7. Conclusion

This paper presented an overview of the CMIR-2025 shared task, aimed at advancing research in Information Retrieval for Bengali–English code-mixed social media data. Unlike the previous edition (CMIR-2024) which focused solely on Roman-script content, this year's task increased complexity by retaining Bengali words in their native script, resulting in a mixed-script retrieval setup that better reflects real-world usage.

A total of eight teams participated, exploring diverse retrieval strategies ranging from classical probabilistic ranking to neural encoders and rank-fusion pipelines. Experimental results show that hybrid and fusion-based systems achieved the best performance across all evaluation metrics, highlighting the importance of combining lexical and semantic signals for robust retrieval in linguistically noisy environments. Despite encouraging outcomes, the overall MAP remains relatively low, underscoring the difficulty of CMIR and the need for more effective cross-script alignment.

We hope that CMIR-2025 serves as a benchmark for future research on code-mixed retrieval, encouraging the development of scalable models capable of handling informal, multilingual communication. Future directions include expanding query volume, incorporating more languages, providing richer annotations, and exploring LLM-based retrieval mechanisms. We believe this task will continue to inspire progress toward inclusive and linguistically equitable information access.

Declaration on Generative AI

In the course of preparing this manuscript, the author(s) employed the generative AI tool ChatGPT. Its use was limited to performing checks for grammar and spelling. Following this, the author(s) conducted a thorough review and revision of the text and assume full responsibility for the final published content.

Acknowledgment

We would like to express our sincere gratitude to FIRE track organizers for providing us with the opportunity to organize this task as part of FIRE 2025.

References

[1] S. Chanda, A. Misha, S. Pal, Advancing language identification in code-mixed tulu texts: Harnessing deep learning techniques, in: FIRE (Working Notes), 2023, pp. 223–230.

[2] S. Chanda, S. Pal, Irlab@ iitbhu@ dravidian-codemix-fire2020: Sentiment analysis for dravidian languages in code-mixed text, in: FIRE (Working Notes), 2020, pp. 535–540.

[3] S. Chanda, A. Mishra, S. Pal, Sentiment analysis and homophobia detection of code-mixed dravidian languages leveraging pre-trained model and word-level language tag, in: Working Notes of FIRE 2022-Forum for Information Retrieval Evaluation (Hybrid). CEUR, 2022.

[4] S. Chanda, A. Mishra, S. Pal, Sentiment analysis of code-mixed dravidian languages leveraging pretrained model and word-level language tag, Natural Language Processing (2024) 1–23. doi:10.1017/nlp.2024.30.

[5] S. Chanda, S. Sheth, S. Pal, Coarse and fine-grained conversational hate speech and offensive content identification in code-mixed languages using fine-tuned multilingual embedding, in: Forum for Information Retrieval Evaluation (Working Notes)(FIRE). CEUR-WS. org, 2022, pp. 502–512.

[6] S. Chanda, A. Mishra, S. Pal, Sarcasm detection in tamil and malayalam dravidian code-mixed text, in: FIRE (Working Notes), 2023.

[7] A. Deroy, S. Maity, Retrievegpt: Merging prompts and mathematical models for enhanced code-mixed information retrieval, in: FIRE 2024 Working Notesl, CEUR Workshop Proceedings, 2024, p. 129–139. URL: https://ceur-ws.org/Vol-4054/T2-2.pdf.

[8] A. Sharma, Infotextcm: Addressing code-mixed data retrieval challenges via text classification, in: FIRE 2024 Working Notesl, CEUR Workshop Proceedings, 2024, p. 140–147. URL: https://ceur-ws.org/Vol-4054/T2-3.pdf.

[9] G. Salton, A. Wong, C. S. Yang, A vector space model for automatic indexing, Commun. ACM 18 (1975) 613–620. URL: http://doi.acm.org/10.1145/361219.361220. doi:http://doi.acm.org/10.1145/361219.361220.

[10] S. Robertson, H. Zaragoza, The probabilistic relevance framework: Bm25 and beyond, Found. Trends Inf. Retr. 3 (2009) 333–389. URL: https://doi.org/10.1561/1500000019. doi:10.1561/1500000019.

[11] J. M. Ponte, W. B. Croft, A language modeling approach to information retrieval, in: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '98, Association for Computing Machinery, New York, NY, USA, 1998, p. 275–281. URL: https://doi.org/10.1145/290941.291008. doi:10.1145/290941.291008.

[12] C. D. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge University Press, 2008. URL: http://www-csli.stanford.edu/~hinrich/information-retrieval-book.html.

[13] D. Mandal, M. Gupta, S. Dandapat, P. Banerjee, S. Sarkar, Bengali and hindi to english clir evaluation, in: C. Peters, V. Jijkoun, T. Mandl, H. Müller, D. W. Oard, A. Peñas, V. Petras, D. Santos (Eds.), Advances in Multilingual and Multimodal Information Retrieval, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 95–102.

[14] S. Bandyopadhyay, T. Mondal, S. K. Naskar, A. Ekbal, R. Haque, S. R. Godhavarthy, Bengali, hindi and telugu to english ad-hoc bilingual task at clef 2007, in: C. Peters, V. Jijkoun, T. Mandl, H. Müller, D. W. Oard, A. Peñas, V. Petras, D. Santos (Eds.), Advances in Multilingual and Multimodal Information Retrieval, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 88–94.

[15] S. Banerjee, K. Chakma, S. K. Naskar, A. Das, P. Rosso, S. Bandyopadhyay, M. Choudhury, Overview of the mixed script information retrieval (MSIR) at FIRE-2016, in: P. Majumder,

M. Mitra, P. Mehta, J. Sankhavara (Eds.), Text Processing - FIRE 2016 International Workshop, Kolkata, India, December 7-10, 2016, Revised Selected Papers, volume 10478 of *Lecture Notes in Computer Science*, Springer, 2016, pp. 39–49. URL: https://doi.org/10.1007/978-3-319-73606-8_3. doi:10.1007/978-3-319-73606-8_3.

[16] K. Chakma, A. Das, Cmir: A corpus for evaluation of code mixed information retrieval of hindi-english tweets, Computación y Sistemas 20 (2016) 425–434. URL: https://api.semanticscholar.org/CorpusID:11152913.

[17] M. Kowsher, I. Hossen, S. S. Ahmed, Bengali information retrieval system(birs), International Journal on Natural Language Computing (2019). URL: https://api.semanticscholar.org/CorpusID:208284965.

[18] S. Ghosh, S. Ghosh, D. Das, Labeling of query words using conditional random field, CoRR abs/1607.08883 (2016). URL: http://arxiv.org/abs/1607.08883. arXiv:1607.08883.

[19] D. Jain, Da-iict in fire 2015 shared task on mixed script information retrieval, in: Fire, 2015. URL: https://api.semanticscholar.org/CorpusID:2392052.

[20] K. Bali, J. Sharma, M. Choudhury, Y. Vyas, "I am borrowing ya mixing ?" an analysis of English-Hindi code mixing in Facebook, in: M. Diab, J. Hirschberg, P. Fung, T. Solorio (Eds.), Proceedings of the First Workshop on Computational Approaches to Code Switching, Association for Computational Linguistics, Doha, Qatar, 2014, pp. 116–126. URL: https://aclanthology.org/W14-3914/. doi:10.3115/v1/W14-3914.

[21] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, W.-t. Yih, Dense passage retrieval for open-domain question answering, in: B. Webber, T. Cohn, Y. He, Y. Liu (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 6769–6781. URL: https://aclanthology.org/2020.emnlp-main.550/. doi:10.18653/v1/2020.emnlp-main.550.

[22] S. Chanda, S. Pal, The effect of stopword removal on information retrieval for code-mixed data obtained via social media, SN Comput. Sci. 4 (2023). URL: https://doi.org/10.1007/s42979-023-01942-7. doi:10.1007/s42979-023-01942-7.

[23] S. Chanda, S. Pal, Overview of the shared task on code-mixed information retrieval from social media data, in: Proceedings of the 16th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE '24, Association for Computing Machinery, New York, NY, USA, 2025, p. 29–31. URL: https://doi.org/10.1145/3734947.3735670. doi:10.1145/3734947.3735670.

[24] S. Chanda, S. Pal, Overview of the shared task on code-mixed information retrieval from social media data, in: FIRE 2024 Working Notesl, CEUR Workshop Proceedings, 2024, p. 124–128. URL: https://ceur-ws.org/Vol-4054/T2-1.pdf.

[25] K. Chakma, S. Datta, A multi-model lexical fusion approach for english–bengali code-mixed information retrieval, in: FIRE 2025 Working Notes, CEUR Workshop Proceedings, 2025.

[26] R. Nagaraju, H. L. Shashirekha, Model fusion for bridging linguistic variability in bengali-english code-mixed information retrieval, in: FIRE 2025 Working Notes, CEUR Workshop Proceedings, 2025.

[27] S. Gupta, T. Nath, V. Gupta, M. Gupta, Lexisemir: A two-stage re-ranking framework with bm25 and zero-shot bi-encoder, in: FIRE 2025 Working Notes, CEUR Workshop Proceedings, 2025.

[28] L. Sawhney, S. Goswami, From romanised to relevant: Multistage information retrieval for code-mixed multilingual queries, in: FIRE 2025 Working Notes, CEUR Workshop Proceedings, 2025.

[29] B. Merchant, A. Khazi, S. S. Sonawane, Reciprocal rank fusion based hybrid dense–sparse information retrieval on code-mixed banglish social media text, in: FIRE 2025 Working Notes, CEUR Workshop Proceedings, 2025.

[30] A. M. Shetty, A. Hegde, S. Coelho, Code-mixed ir: An ensemble approach to robust ranking models, in: FIRE 2025 Working Notes, CEUR Workshop Proceedings, 2025.

[31] D. Shroff, Colir: Late interaction based code-mixed information retrieval for english-bengali language pair, in: FIRE 2025 Working Notes, CEUR Workshop Proceedings, 2025.

[32] H. Mishra, R. Sharma, N. Yadav, Decoding benglish: Scalable information retrieval for transliterated

code-mixed conversations, in: FIRE 2025 Working Notes, CEUR Workshop Proceedings, 2025.