Code-Mixed IR: An Ensemble Approach to Robust Ranking Models

Amrithkala M Shetty[1], Asha Hegde[2] and Sharal Coelho[2]

[1]Department of Computer Applications, Nitte Institute of Professional Education, Nitte (Deemed to be University), Karnataka, India
[2]Department of Computer Science, Mangalore University, India

Abstract

The proliferation of code-mixed languages, such as Bengali-English in Roman-transliterated form, on digital platforms poses significant challenges for information retrieval systems, particularly in handling natural language questions for document-level retrieval. This study addresses the task of developing an IR system capable of retrieving relevant documents from a corpus of 107,900 documents in response to unseen Bengali-English code-mixed queries. In this paper, we – team NLPFusion – describe our proposed model submitted to the CMIR-2025 shared task, which 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. The models performance is evaluated on a test set of 30 queries, the system achieved a MAP score of 0.136089, NDCG score of 0.319129, P@5 of 0.286667, and P@10 of 0.23. The results highlight the efficacy of ensemble methods in addressing the linguistic complexity of code-mixed IR, offering a scalable solution for multilingual search applications in diverse linguistic contexts.

Keywords

Information Retrieval, Code-Mixed, Ranking models

1. Introduction

In recent years, multilingual and code-mixed material has become much more common on blogs, social media, and other digital platforms. Particularly in multilingual societies like India, the increasing growth of multilingual and code-mixed content on digital platforms presents difficult issues for Information Retrieval (IR) and Natural Language Processing (NLP) [1]. Code-mixing is the practice of speakers switching between two or more languages [2] [3]. It can occur at various linguistic levels, including the word or sub-word, sentence where users blend their native and/or local languages [4].

As online social networks continue to grow, many of its users communicate in native languages using foreign scripts. One notable trend is the use of the Roman script to communicate in native languages on social media platforms. This is a norm in India, where people combine English and their native languages in their social media posts [5]. Social media platforms are becoming more popular. This results in an enormous amount of user-generated content. It is becoming increasingly difficult and impractical to handle and interpret all that text manually. Information retrieval [6], automatic speech recognition, Machine Translation (MT)[7], language identification, POS tagging, and sentiment analysis [8] for Indian languages are a few prominent uses of code-mixing.

Traditional Information Retrieval (IR) systems mainly designed for monolingual datasets, face challenges when dealing with the complexities of code-mixed data. Research on IR for code-mixed languages still needs to be improved, despite the many developments in multilingual NLP. Language identification, hate speech detection, and sign of depression have accounted for a large portion of current research. However, their use in IR in languages with limited resources, such as Bengali, has to be improved.

The Bengali-English code mixing poses unique challenges for IR due to the inherent linguistic differences between the two languages [9]. Code-Mixed Information Retrieval (CMIR) [10] is challenging because queries written either in native or Roman scripts need to be matched to the documents written in either or both the scripts [5]. This paper aims to explore and propose solutions for the task of IR in

Forum for Information Retrieval Evaluation, December 17-20, 2025, India
amrithkalas@gmail.com (A. M. Shetty); hegdekasha@gmail.com (A. Hegde); sharalmucs@gmail.com (S. Coelho)

code-mixed data. The aim is to develop a mechanism to pinpoint the most relevant answers from these code-mixed conversations. The focus is on Roman transliterated Bengali mixed with English language.

The remainder of this paper is organized as follows: Section 2 reviews the related work. Section 3 contains the Task description and Section 4 describes the proposed methodology. Section 5 outlines the experiments and result. Finally, Section 5 concludes the paper.

2. Related Work

In recent years, researchers have shown growing interest in code-mixed text, particularly in low-resource languages, for a variety of applications such as MT, language identification, POS tagging and IR [11]. There have been several studies on code-mixed data, cross lingual IR [12] [13] and on multi lingual IR including Indian languages. To work with language identification task in code-mixed Bengali-English and Hindi-English text, Mandal and Singh [14] developed a multichannel neural network model of CNN and LSTM models combined with Bidirectional LSTM and Conditional Random Fields. This multichannel NN model achieved accuracies of 93.32% and 93.28% for Hindi-English and Bengali-English data, respectively. Supriya and Sukomal [9] demonstrated the effect of stopword removal on IR for code-mixed text collected from social media. In their study, they obtained empirical evaluations that a significant progress in MAP occurred when stopwords were eliminated compared to cases where they were not. Recent studies have explored IR for code-mixed languages, focusing on low-resource and Indic language contexts. Bali et al.[15] explored transliteration normalization for Hindi-English code-mixed text. They developed algorithms to map Romanized text back to its original script, enabling more accurate processing by traditional NLP models. Chakma and Das [5] specifically addressed the consequences of IR for Code-Mixed social media texts, mostly from data collected via Twitter. They collected Hindi-English tweets about the popular events. Twenty questions with an average query length of 2.67 phrases were subjected to the Boolean model by the author. A MAP value of 0.186 was attained.

While these studies provide valuable insights into code-mixing, transliteration, and information retrieval, there is a noticeable gap in addressing the specific challenges of extracting relevant information from code-mixed conversations in Roman transliterated Bengali.

3. Task description

The task is to develop retrieval systems that return documents in a specific Bengali-English code-mixed language when given a query in the same code-mixed language[1]. The aim here is to create an IR model using artificial intelligence that are capable of handling previously unseen queries effectively. The retrieval process is at the document level, with queries presented as natural language questions. Documents are considered relevant if they contain answers to these questions.

4. Methodology

This research utilizes an IR pipeline with PyTerrier for ranking document evaluation and aggregation of several retrieval models to rank documents against queries. The experimental methodology includes the indexing of a document collection, document retrieval with various models, ensemble fusion method application, and performance measurement on a validation set, followed by submission generation on the test set.

4.1. Retrieval Models

Three retrieval models are utilized in the proposed methodology and each one is described below:

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

- BM25: A probabilistic model that ranks documents based on term frequency and inverse document frequency, implemented via PyTerrier's BatchRetrieve with the BM25 (Best Matching 25) weighting model.
- PL2: A divergence-from-randomness model that uses a Poisson approximation for term frequency, also implemented via BatchRetrieve with the PL2 weighting model.
- Hiemstra Language Model (Hiemstra_LM): A language model incorporating Dirichlet smoothing for document ranking, implemented via BatchRetrieve.

Each model retrieves ranked lists of documents for the training and test query sets, producing dataframes with columns qid (query ID), docno (document ID), score (relevance score), and rank (document rank).

4.2. Ensemble Fusion Techniques

To improve retrieval performance, three ensemble fusion methods are applied to combine the results from BM25, PL2, and Hiemstra_LM:

- CombSUM: This method merges the ranked lists by summing the normalized scores of documents across the three models. Documents are re-ranked based on the aggregated scores, with ties resolved using dense ranking.
- CombMNZ: An extension of CombSUM, this method multiplies the summed scores by the number of non-zero scores a document receives across the models, rewarding documents retrieved by multiple systems. The final ranking is based on these adjusted scores.
- Reciprocal Rank Fusion (RRF): This method computes a score for each document based on the reciprocal of its rank (1/(k + rank)) in each model's ranked list, where k=60 is a constant to stabilize the fusion. Scores are summed across models, and documents are ranked based on the aggregated RRF scores.

5. Experiments and Result

The experiments are performed using the given dataset for the Bengali-English CMIR-2025 shared task. The corpus contains 107,900 documents in the training set, out of which 20 code-mixed queries/topics in Roman-transliterated Bengali mixed with English are available for training, and 30 queries in the test set. Three ranking models are created and merged using ensemble fusion techniques: CombSUM, CombMNZ, and Reciprocal Rank Fusion (RRF). CombSUM combines normalized document scores from models and ranks based on total scores, breaking ties by dense ranking. CombMNZ generalizes CombSUM by multiplying total scores by the number of non-zero scores, favoring documents retrieved by several models. RRF calculates scores by the reciprocal of rank (1/(k + rank), k=60) and combines them for final ranking.

To ensure fair and consistent evaluation, the models are all tested using standard IR measures such as Mean Average Precision (MAP), Normalized Discounted Cumulative Gain (NDCG), Precision at 5 (P@5), and Precision at 10 (P@10).

5.1. Results

The performance of five submission runs is summarized in the Table 1, based on the evaluation metrics for the test set of 30 queries [16, 17]. Runs 1, 2, and 3 achieved similar performance, with a MAP score of 0.136089, NDCG score of 0.319129, P@5 of 0.286667, and P@10 of 0.23, indicating consistent performance across these submissions, likely due to similar configurations of the CombSUM or RRF fusion methods. Runs 4 and 5, with lower scores (MAP: 0.092505, NDCG: 0.246699, P@5: 0.233333, P@10: 0.17), suggest the use of a less effective fusion strategy. The results demonstrate that the ensemble approach, particularly in Runs 1-3, effectively retrieved relevant documents for code-mixed

Table 1
Comparison of MAP, NDCG, P@5, and P@10 Scores along with submissions

Submission	MAP Score	ndcg Score	p@5 Score	p@10 Score
Run 1	0.136089	0.319129	0.286667	0.23
Run 2	0.136089	0.319129	0.286667	0.23
Run 3	0.136089	0.319129	0.286667	0.23
Run 4	0.092505	0.246699	0.233333	0.17
Run 5	0.092505	0.246699	0.233333	0.17

queries, with NDCG scores reflecting reasonable ranking quality. However, the moderate MAP and precision scores indicate challenges in handling the linguistic complexity and semantic ambiguity of Roman-transliterated Bengali-English code-mixed text.

6. Conclusion

Code-mixed IR in English and Bengali involves retrieving useful information from a dataset whose content is a mixture of both languages. This is particularly common in multilingual societies where individuals can switch between languages, sometimes even in the middle of a phrase. The developed information retrieval pipeline successfully combined three different retrieval models—BM25, PL2, and Hiemstra Language Model—via PyTerrier to rank documents for specific queries. Through the use of ensemble fusion methods, i.e., CombSUM, CombMNZ, and Reciprocal Rank Fusion (RRF), the system was able to successfully merge strengths from individual models to improve retrieval. Testing on the training set with measures of Mean Average Precision (MAP), Normalized Discounted Cumulative Gain (nDCG) gave an indication of the relative performance of individual and ensemble approaches. The CombSUM fusion strategy was used for creating the final submission on the test set, returning a ranked list of top 100 documents per query. This strategy illustrates the promise of ensemble methods for enhancing retrieval accuracy and insensitivity, providing a scalable solution for managing heterogeneous query sets in information retrieval applications. Tying fusion parameters more optimally or investigating other models might further boost performance in the next iteration.

Declaration on Generative AI

In the preparation of this manuscript, we employed a generative AI assistant in a limited capacity to assist with the writing process. The author(s) utilized Grok[2] for grammar and spelling checks. Paraphrasing was handled via QuillBot. With this tool, the author(s) reviewed and revised the content as required, while assuming full responsibility for the publication's integrity.

References

[1] 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, 2024, pp. 29–31.

[2] S. N. Bhattu, S. K. Nunna, D. V. Somayajulu, B. Pradhan, Improving code-mixed pos tagging using code-mixed embeddings, ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP) 19 (2020) 1–31.

[3] S. Thara, P. Poornachandran, Code-mixing: A brief survey, in: 2018 International conference on advances in computing, communications and informatics (ICACCI), IEEE, 2018, pp. 2382–2388.

[2] https://grok.com

[4] A. Hegde, F. Balouchzahi, S. Coelho, H. Shashirekha, H. A. Nayel, S. Butt, Overview of coli-tunglish: Word-level language identification in code-mixed tulu text at fire 2023., in: FIRE (Working Notes), 2023, pp. 179–190.
[5] 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.
[6] M. N. Moreno-García, Information retrieval and social media mining, 2020.
[7] A. Hegde, H. L. Shashirekha, A. K. Madasamy, B. R. Chakravarthi, A study of machine translation models for kannada-tulu, in: Congress on Intelligent Systems, Springer, 2022, pp. 145–161.
[8] A. M. Shetty, M. F. Aljunid, D. Manjaiah, Sentiment exploring on feedback of e-commerce data using machine learning algorithms, in: International Conference on Emerging Research in Computing, Information, Communication and Applications, Springer, 2023, pp. 107–129.
[9] S. Chanda, S. Pal, The effect of stopword removal on information retrieval for code-mixed data obtained via social media, SN Computer Science 4 (2023) 494.
[10] 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.
[11] Z. Liu, Y. Zhou, Y. Zhu, J. Lian, C. Li, Z. Dou, D. Lian, J.-Y. Nie, Information retrieval meets large language models, in: Companion Proceedings of the ACM Web Conference 2024, 2024, pp. 1586–1589.
[12] P. Bhattacharya, P. Goyal, S. Sarkar, Using communities of words derived from multilingual word vectors for cross-language information retrieval in indian languages, ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP) 18 (2018) 1–27.
[13] V. K. Sharma, N. Mittal, Cross lingual information retrieval (clir): Review of tools, challenges and translation approaches, in: Information Systems Design and Intelligent Applications: Proceedings of Third International Conference INDIA 2016, Volume 1, Springer, 2016, pp. 699–708.
[14] S. Mandal, A. K. Singh, Language identification in code-mixed data using multichannel neural networks and context capture, arXiv preprint arXiv:1808.07118 (2018).
[15] K. Bali, J. Sharma, M. Choudhury, Y. Vyas, "i am borrowing ya mixing?" an analysis of english-hindi code mixing in facebook, in: Proceedings of the first workshop on computational approaches to code switching, 2014, pp. 116–126.
[16] S. Chanda, K. Tewari, S. Pal, Overview of the cmir track at fire 2025: Code-mixed information retrieval from social media data, in: FIRE '25: Proceedings of the 17th Annual Meeting of the Forum for Information Retrieval Evaluation. December 17-20, Varanasi, India, Association for Computing Machinery (ACM), New York, NY, USA, 2025.
[17] S. Chanda, K. Tewari, S. Pal, Findings of the code-mixed information retrieval from social media data (cmir) shared task at fire 2025, in: K. Ghosh, T. Mandl, S. Pal, S. Majumdar, A. Chakraborty (Eds.), Forum for Information Retrieval Evaluation (Working Notes) (FIRE 2025) December 17-20, Varanasi, India, CEUR-WS.org, 2025.