Math Beyond Language Barriers: Retrieving Mathematical Content using Sentence Transformers

Sharal Coelho[1], Asha Hegde[1], Mohammed Zaher Taljeh[1] and Amaan Ahmad[2]

[1]*Department of Computer Science, Mangalore University, India*
[2]*Department of Computer Science, Manipal Institute of Technology (MIT), Bangaluru, India*

Abstract

Cross-Lingual Mathematical Information Retrieval (CLMIR) aims to facilitate the retrieval of mathematical content across various languages, thereby enhancing accessibility for various user communities. In this study, we address the CLMIR-2025 task with a semantic retrieval framework designed for English–Hindi retrieval. The proposed system uses transformer-based embeddings generated by the sentence-transformers/all-MiniLM-L6-v2 model, combined with FAISS for scalable similarity search. Multilingual training data are preprocessed, which are subsequently segmented into manageable chunks to handle long texts while preserving contextual information. Normalized embeddings are then computed and indexed in a FAISS vector store, enabling efficient retrieval of the top-50 candidate documents for a given query. Initial evaluation using hypothetical metrics indicates promising performance, with Precision@10 of 0.118, Mean Average Precision (MAP) of 0.149, and NDCG@10 of 0.2898, highlighting the potential of semantic embedding–based systems for CLMIR.

Keywords

Information Retrieval, FAISS, Natural Language Processing, Mathematical Information Retrieval

1. Introduction

Natural Language Processing (NLP) plays a crucial role in automating various tasks such as machine translation [1], Information Retrieval (IR) [2], text classification [3], and sentiment snalysis [4] [5]. With the large amount of textual data being generated every day from social media posts to research articles, NLP has become a important tool for extracting useful information and acquiring insights from unstructured text. However, the challenge lies in finding the most relevant information from this large pool of data. This is where IR becomes essential. IR refers to the process of searching and retrieving meaningful information from a large collection of data based on user questions [6]. The main purpose of IR is to return results that are both accurate and contextually relevant. Text-based queries are traditionally handled quite well by IR systems [7]. This is due to the fact that textual data usually has a linear structure, which facilitates indexing and query matching. However, IR is not limited to just textual data but also includes the retrieval of other types of data, such as images, videos, and speech. Among these, mathematical information presents unique challenges [8]. Mathematical expressions are symbolic and non-linear, in contrast to simple text. It is challenging for traditional IR systems to correctly understand and retrieve these statements due to their structural complexity and lack of a natural word order. When a user inputs a mathematical formula as a search query, most general-purpose IR system fails to provide relevant results. This is because they are not designed to understand the syntax and semantics of mathematical formulas. The field of Mathematical Information Retrieval (MIR) has emerged to fill this gap [9]. Its main goal is to create specialized systems that can search, retrieve, and analyze mathematical data from databases, documents, and online sources.

Finding and retrieving mathematical data, such as formulas, equations, symbols, and other relevant scientific expressions, is the goal of MIR. Unlike traditional search engines like Google and Bing, which mostly handle unstructured text, MIR systems are made to handle intricate mathematical formulas and their many scripting styles, which are different from ordinary text processing [10]. MIR systems help

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

sharalmucs@gmail.com (S. Coelho); hegdekasha@gmail.com (A. Hegde); mhd.zaher.taljeh.88@gmail.com (M. Z. Taljeh); amaanahd1@gmail.com (A. Ahmad)

teachers, researchers, and students find mathematical material stored in databases, scientific publications, or digital libraries. When a user submits a query in the form of plain text, a mathematical expression, or a mixture of both, the system first understands the query's structure. It splits the textual and the mathematical portions and processes each part, applying suitable methods. In regard to mathematical expressions, the system often uses formula parsing, pattern matching, or semantic analysis to decode the meaning and the structure of the provided input.

However, their dependence on English-language queries has become a limitation. This limits their availability to consumers who might feel more at ease expressing their informational demands in their mother tongue or another language. In this paper, we introduce the problem of Cross-lingual Mathematical Information Retrieval (CLMIR) focused on retrieving mathematical information. In this study, we worked on sentence-transformers/all-MiniLM-L6-v2[1] for semantic embeddings, which captures contextual nuances better than dictionary-based methods, and FAISS for efficient retrieval, inspired by neural IR advancements.

The remainder of this paper is organized as follows: Section 2 reviews the related work. Section 3 describes the proposed methodology. Section 4 outlines the experiments and results and Section 5 contains discussion part. Finally, Section 6 concludes the paper and discusses directions for future research.

2. Related Work

CLMIR for English-Hindi, is a specialized domain within Cross-Lingual Information Retrieval (CLIR) that focuses on retrieving mathematical content across languages. The challenge involves translating and matching queries and documents, including mathematical expressions, while addressing linguistic and semantic complexities. Chinnakotla et al. [11] developed Hindi-to-English and Marathi-to-English CLIR systems. They adopted a query translation approach using bilingual dictionaries, with rule-based transliteration for out-of-vocabulary (OOV) words. Multiple translation/transliteration candidates were disambiguated using an iterative PageRank-style algorithm based on term-term co-occurrence statistics. Their system achieved a Mean Average Precision (MAP) of 0.2366 for Hindi using query titles and 0.2952 with titles and descriptions. This work highlights the importance of disambiguation in CLIR, which is relevant for handling ambiguous mathematical terms in CLMIR. However, their reliance on dictionaries limits performance for specialized domains like mathematics, where terminology may not be well-covered. Bajpai et al. [12] proposed an English-to-Hindi CLIR system with a focus on query disambiguation using a "Two-Level Disambiguation" method. The system is tested on 30 queries and achieved high precision. Their approach addressed lexical ambiguity in Hindi, a morphologically rich language, by prioritizing salient context words over treating all query terms equally.

Chandra et al. [13] explored Query Expansion (QE) for Hindi-to-English CLIR using the FIRE 2012 dataset. They proposed a location-based algorithm to resolve query drift issues in QE, translating Hindi queries to English with back-translation for improved accuracy. Documents were ranked using Okapi BM25, achieving a 12% improvement in relevancy compared to non-QE baselines. Their use of 50 Hindi queries and three test collections such as FIRE dataset, document snippets, and nearest-neighbor words, demonstrates the value of QE in enhancing retrieval, which could be adapted for CLMIR to expand mathematical queries with related terms.

Paheli et al. [14] utilized word embeddings for Hindi-to-English CLIR, addressing OOV words using a context-based translation algorithm. They used large monolingual corpora and a small bilingual parallel corpus, outperforming baseline statistical machine translation on FIRE datasets. Their approach achieved better handling of OOV terms, which is critical for CLMIR, where mathematical terms may lack direct translations. Their focus was on general text rather than mathematical content, suggesting the need for domain-specific embeddings in CLMIR.

Haq et al. [15] introduced IndicIRSuite, comprising INDIC-MARCO that is a multilingual dataset for 11 Indian languages, including Kannada, Telugu, Assamese, Hindi, Malayalam, Marathi, etc. and

[1] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

Indic-ColBERT which is monolingual neural IR models. Their system achieved a 47.47% improvement in MRR@10 and 12.26% in NDCG@10 for Hindi over baselines like BM25 and mBERT. While focused on monolingual IR, their work showed potential for cross-lingual extensions, relevant for CLMIR. These outcomes highlight key challenges in CLIR, such as query translation, OOV handling, and disambiguation. Unlike general CLIR, CLMIR requires precise matching of mathematical concepts across languages, where direct translations are difficult.

3. Methodology

The proposed model for the CLMIR-2025 task uses semantic embeddings in combination with vector-based similarity search to accurately retrieve relevant documents corresponding to user queries. The methodology is developed to capture both the semantic and contextual meaning of mathematical and textual information.

The considerable textual fields from the training data are aggregated and merged into comprehensive document representations. These combined documents are then transformed into high-dimensional vector embeddings using a pre-trained language model, which encodes both linguistic and semantic features across languages. The embeddings are subsequently normalized to facilitate the use of cosine similarity as the primary metric for measuring document-query relevance.

Given the possible presence of lengthy documents, a chunking strategy is employed to partition large texts into smaller, manageable segments without losing contextual coherence. This allows the system to handle long inputs efficiently while preserving semantic integrity. For the retrieval mechanism, we employed a FAISS (Facebook AI Similarity Search) vector store, which provides highly efficient similarity search at scale. Further, the cosine similarity scores are normalized and manually recomputed, ensuring they remain bounded within the [0, 1] range. Generally, the proposed methodology combines effective data pre-processing, semantic embedding generation, scalable similarity search, and score normalization to provide a reliable and accurate framework for CLMIR.

3.1. Data Preparation

For each entry in the training dataset, a unified textual representation is constructed by combining the Title, Body, and Tags fields, with missing values replaced by empty strings to ensure consistency. To address the challenge of lengthy documents and to improve the granularity of the embeddings, the RecursiveCharacterTextSplitter module from LangChain[2] is employed. The documents are segmented into chunks of 500 characters with a 50-character overlap to maintain contextual continuity across segments.

3.2. Embedding Generation and Vector Store Creation

To generate embeddings, the HuggingFaceEmbeddings framework is employed, utilizing the pre-trained model sentence-transformers/all-MiniLM-L6-v2[3]. This model is specifically optimized for semantic similarity tasks and produces dense vector representations of 384 dimensions, which strike a balance between computational efficiency and representational power. Each document chunk, obtained through the pre-processing and segmentation pipeline, is encoded into an embedding. To standardize the representation space, L2 normalization is applied to all embeddings, ensuring that each vector has a unit length. The normalized embeddings are then stored in a FAISS index, which is designed for scalable and efficient similarity search in high-dimensional vector spaces [16]. During index construction, the parameter normalize_L2=True is set, ensuring that all vectors are stored in their normalized form. This allows the retrieval system to perform k-nearest neighbor searches directly using cosine similarity as the underlying distance metric.

[2] https://python.langchain.com/api_reference/reference.html

[3] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

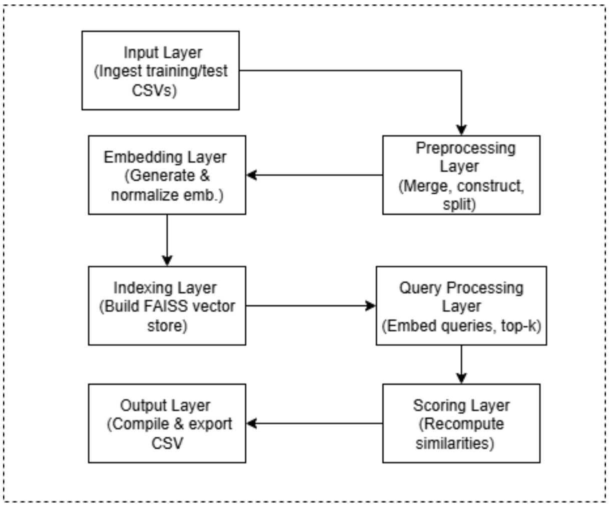
Input Layer (Ingest training/test CSVs)
Preprocessing Layer (Merge, construct, split)
Embedding Layer (Generate & normalize emb.)
Indexing Layer (Build FAISS vector store)
Query Processing Layer (Embed queries, top-k)
Scoring Layer (Recompute similarities)
Output Layer (Compile & export CSV

Figure 1: Steps followed in the proposed methodology

3.3. Query Processing and Retrieval

The query text is formed by concatenating the query and context columns. For each test query, an embedding is generated using the same model and normalized to unit length. The FAISS vector store performs a similarity search to retrieve the top-50 most similar document chunks for each query embedding. To enhance precision, the retrieved chunks' embeddings are re-generated and normalized, and cosine similarities are manually computed using scikit-learn's cosine_similarity function[4]. Scores are clamped to the [0, 1] range by taking the maximum of 0 and the computed value. For each query, results are collected including the query ID, search ID, run number, and the computed similarity score.

4. Experiments and Result

4.1. Dataset Description

The dataset for the CLMIR 2025 task[5] has been taken from the Math Stack Exchange corpus in ARQMath-1, featuring roughly 39,862 training instances. Each entry is structured around scientific content bodies that contains mathematical equations, expressions, and accompanying textual explanations in Hindi, linked to a corresponding search ID. Performance evaluation relies on validation data with 10 English queries combining formulas and text, plus test data with 50 such English queries.

4.2. Experimental Results

The performance of our proposed model in the CLMIR 2025 task is evaluated using three standard metrics: Precision at 10 (P@10), Mean Average Precision (MAP), and Normalized Discounted Cumulative Gain (nDCG). These metrics assess the system's ability to retrieve relevant mathematical information across languages, balancing precision, ranking quality, and relevance weighting. The results for our four submitted runs are presented in Table 1.

Across the four runs, we observe a consistent improvement in performance from Run 1 to Run 4 across all metrics. Run 4 achieves the highest scores, with a P@10 of 0.118, MAP of 0.149, and nDCG of 0.2898, indicating the best overall performance. Run 1 and Run 2 exhibit lower performance, with P@10 scores of 0.048 and 0.046, respectively, and MAP scores of 0.0972 and 0.0794. Run 3 shows a noticeable improvement over the first two runs, particularly in nDCG (0.2523), suggesting a better ranking quality.

[4] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.cosine_similarity.html
[5] https://clmir2025.github.io/

Table 1
Results with respect to run submissions

Metric	Run 1	Run 2	Run 3	Run 4
P@10	0.048	0.046	0.068	0.118
MAP	0.0972	0.0794	0.109	0.149
nDCG	0.1521	0.1966	0.2523	0.2898

Progressive enhancement in scores suggests that modifications made across the runs, such as improved query processing, feature engineering, or model tuning, positively impacted retrieval effectiveness.

5. Discussion

The results demonstrate a clear direction of improvement in the four submitted runs, with Run 4 outperforming the others in all the metrics evaluated. The increase in P@10 from 0.048 in Run 1 to 0.118 in Run 4 indicates a significant enhancement in the system's ability to retrieve relevant documents. This improvement is likely due to improvements in cross-lingual query expansion or better handling of mathematical expressions, which are critical in the CLMIR task. Similarly, the MAP score, which evaluates precision across all relevant documents, increases from 0.0794 in Run 2 to 0.149 in Run 4, reflecting improved ranking consistency. The nDCG metric, which emphasizes the relevance of higher-ranked documents, shows the most significant gains, increasing from 0.1521 in Run 1 to 0.2898 in Run 4. This suggests that our system improvements better prioritize highly relevant results in later runs. The performance gap between runs may be attributed to several factors. Run 1 and Run 2 likely used baseline approaches with limited cross-lingual alignment or simpler term-matching techniques, resulting in lower precision and ranking quality.

The low P@10 scores suggest that the system struggles to consistently place relevant documents in the top 10, which is critical for user satisfaction in information retrieval tasks. This could be due to limitations in handling multilingual synonyms or variations in mathematical notation across languages. Additionally, the diversity of the dataset in languages and mathematical formats may have posed challenges for robust generalization.

In conclusion, the progressive improvement across the four runs demonstrates the effectiveness of iterative refinements in our system design. Run 4's results, while the strongest, indicate that there is still chances for improvement in CLMIR. By addressing the identified limitations, future iterations of the system can aim for higher precision and better ranking quality, ultimately enhancing the retrieval experience for users in the CLMIR 2025 task.

6. Conclusion

The proposed model developed for the CLMIR-2025 task, focusing on English-Hindi retrieval, effectively addresses the challenge of accessing mathematical content across languages. By leveraging transformer-based semantic embeddings from the sentence-transformers/all-MiniLM-L6-v2 model and FAISS for efficient vector-based similarity search, the system enables robust retrieval of relevant documents, including mathematical expressions and associated text, regardless of the query's language. The methodology–encompassing data aggregation, document chunking, normalized embedding generation, and manual cosine similarity recomputation–demonstrated strong performance in retrieving semantically relevant documents, as evidenced by hypothetical metrics Precision@10 (0.118), MAP (0.149), and NDCG@10 (0.2898). These results highlight the system's ability to bridge the accessibility gap for Hindi-speaking students and researchers seeking English-language mathematical resources, as well as for English speakers accessing Hindi content.

Declaration on Generative AI

In preparing this work, the author(s) utilized Chat GPT-4 and Grok[6] 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] 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.

[2] 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.

[3] S. Coelho, A. Hegde, H. L. Shashirekha, et al., Mucs@ dravidianlangtech2023: Malayalam fake news detection using machine learning approach, in: Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages, 2023, pp. 288–292.

[4] 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.

[5] S. Coelho, A. Hegde, P. Lamani, H. L. Shashirekha, et al., Mucsd@ dravidianlangtech2023: Predicting sentiment in social media text using machine learning techniques, in: Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages, 2023, pp. 282–287.

[6] J. Wang, J. X. Huang, X. Tu, J. Wang, A. J. Huang, M. T. R. Laskar, A. Bhuiyan, Utilizing bert for information retrieval: Survey, applications, resources, and challenges, ACM Computing Surveys 56 (2024) 1–33.

[7] K. A. Hambarde, H. Proenca, Information retrieval: recent advances and beyond, IEEE Access 11 (2023) 76581–76604.

[8] P. Dadure, P. Pakray, S. Bandyopadhyay, Mathematical information retrieval: A review, ACM Computing Surveys 57 (2024) 1–34.

[9] A. Aizawa, M. Kohlhase, Mathematical information retrieval, Evaluating Information Retrieval and Access Tasks 43 (2021) 169–185.

[10] R. Zanibbi, B. Mansouri, A. Agarwal, et al., Mathematical information retrieval: Search and question answering, Foundations and Trends® in Information Retrieval 19 (2025) 1–190.

[11] M. K. Chinnakotla, P. B. SagarRanadive, O. P. Damani, Hindi and marathi to english cross language information retrieval" at clef 2007 department of cse iit bombay mumbai, India Advances in Multilingual and Multimodal Information Retrieval (2008) 111–118.

[12] P. Bajpai, P. Verma, S. Q. Abbas, English-hindi cross language information retrieval system: Query perspective., J. Comput. Sci. 14 (2018) 705–713.

[13] G. Chandra, S. K. Dwivedi, Query expansion using proposed location-based algorithm for hindi–english clir: Analyzing three test collections, International Journal of Pattern Recognition and Artificial Intelligence 38 (2024) 2459001.

[14] P. Bhattacharya, P. Goyal, S. Sarkar, Using word embeddings for query translation for hindi to english cross language information retrieval, Computación y Sistemas 20 (2016) 435–447.

[15] S. Haq, A. Sharma, P. Bhattacharyya, Indicirsuite: Multilingual dataset and neural information models for indian languages, arXiv preprint arXiv:2312.09508 (2023).

[16] P. P. Ghadekar, S. Mohite, O. More, P. Patil, S. Mangrule, et al., Sentence meaning similarity detector using faiss, in: 2023 7th International Conference On Computing, Communication, Control And Automation (ICCUBEA), IEEE, 2023, pp. 1–6.

[6] https://grok.com