MIRaCLE: Multilingual Information Retrieval with Cross-Lingual Embeddings for Mathematical Expressions

Krishna Tewari[1,*], Supriya Chanda[2] and Riti Tripathi[3]

[1]Indian Institute of Technology (BHU), Varanasi, INDIA
[2]Bennett University, Greater Noida, INDIA
[3]Kalinga Institute of Industrial Technology, Bhubaneswar, INDIA

Abstract

Cross-Lingual Mathematical Information Retrieval (CLMIR) addresses the challenge of retrieving documents containing mathematical expressions and text across languages, an area where most existing systems remain monolingual. As part of the FIRE 2025 CLMIR shared task, which targets English-Hindi retrieval using a curated corpus of 39,862 Hindi instances from Math StackExchange (ARQMath-1) and 50 English queries, we developed and evaluated two hybrid retrieval models. Run 01 combines the multi-qa-MiniLM-L6-cos-v1 model with regex-based text cleaning, while Run 02 leverages the all-mpnet-base-v2 model, spaCy-based preprocessing, and an enhanced numerical similarity function for mathematical expressions. Both systems adopt a hybrid scoring strategy integrating semantic embeddings and symbolic math similarity, with FAISS employed for efficient large-scale indexing. Performance was assessed using Precision@10 (P@10), Mean Average Precision (MAP), and normalized Discounted Cumulative Gain (nDCG). Our Run 2 achieved competitive results (P@10: 0.122, MAP: 0.165, nDCG: 0.3063), ranking among the top-performing teams in CLMIR 2025. The results underscore the effectiveness of robust multilingual embeddings and refined math similarity computation, while suggesting future improvements through adaptive weighting and multi-expression handling.

Keywords

Cross-Lingual Information Retrieval, Mathematical Information Retrieval, Sentence Transformers, FAISS, SymPy

1. Introduction

The increasing availability of multilingual scientific and educational content has amplified the need for robust *Cross-Lingual Mathematical Information Retrieval (CLMIR)* systems. At its core, CLMIR seeks to retrieve documents containing both mathematical expressions and text across different languages. A query $Q = (q_t^1, q_t^2, \ldots, q_t^T)$ may thus consist of a sequence of natural language tokens interleaved with symbolic mathematical components, and the retrieval system must align this composite query against a heterogeneous collection of documents. For example, in the query *"What is the value of* $\int e^x dx$*?"* issued in English, the system should be able to retrieve relevant solutions written in Hindi that correctly state $\int e^x dx = e^x + C$.

The cross-lingual mathematical retrieval setting poses distinctive challenges. First, **linguistic diversity**, where mathematical terminology varies across languages (e.g., "integration" in English vs. "समाकलन'' in Hindi). Second, **structural variation**, since equivalent formulae can be expressed in different notational or syntactic forms, complicating symbolic matching. Third, **semantic alignment**, as effective retrieval requires joint reasoning over text and mathematical structure across languages. These issues are compounded by the inherent imbalance in multilingual datasets, where English dominates most scientific discourse while non-English resources, such as Hindi, remain underrepresented.

To advance research in this direction, the Forum for Information Retrieval Evaluation (FIRE) has introduced the CLMIR shared task in 2025, focusing on English-Hindi retrieval. The dataset, derived

Forum for Information Retrieval Evaluation, December 17-20, 2025, Varanasi, India
*Corresponding author.
krishnatewari.rs.cse24@iitbhu.ac.in (K. Tewari); suplife24@gmail.com (S. Chanda); rititripathi09@gmail.com (R. Tripathi)
0009-0005-6599-9956 (K. Tewari); 0000-0002-6344-8772 (S. Chanda)

from Math Stack Exchange, contains approximately 39,862 Hindi instances, enriched with both formulae and explanatory text, and provides 50 English queries for system evaluation.

In this work, we present our participation in the FIRE 2025 CLMIR shared task. We propose a hybrid architecture that integrates multilingual sentence-transformer embeddings with SymPy-based formula similarity measures, and employs FAISS for efficient large-scale indexing. To enhance robustness, we adopt preprocessing strategies for handling noisy mathematical notation and employ hybrid scoring functions that balance semantic and symbolic similarity. Evaluation on the official benchmark demonstrates competitive performance, underscoring the effectiveness of combining multilingual neural representations with symbolic reasoning for CLMIR.

The rest of this paper is organized as follows: Section 2 discusses related work; Section 3 describes the dataset; Section 4 presents our proposed methodology; Section 5 reports results and analysis; and Section 6 concludes with key findings and future directions.

2. Related Work

Research in *Mathematical Information Retrieval (MIR)* has addressed the problem of representing mathematical content in ways that allow both symbolic and textual aspects to be effectively indexed and retrieved. A major focus has been on developing structural encodings of formulae that preserve their semantics while enabling retrieval operations similar to text search. Canonicalization and operator tree methods provided one of the earliest means to support structural similarity, showing improvements in identifying related expressions beyond surface string matching [1, 2].

The question of how to index mathematical expressions efficiently has been widely studied. Early indexing systems designed specifically for formulae applied tree-based and string-based encodings, demonstrating scalable retrieval across large digital libraries [3, 4]. Further refinements incorporated text metadata and ranking strategies, where structural signals were combined with contextual information to improve relevance estimation [5, 6].

Benchmark-driven research provided a strong impetus for MIR development. The *NTCIR-11 Math-2 Task* introduced a formula retrieval evaluation using a Wikipedia corpus of 100 test queries, with metrics such as Precision@5 and Mean Reciprocal Rank (MRR) [7]. Participating systems demonstrated that hybrid approaches combining formula encodings with semantic text improved retrieval. For instance, the IFISB system extracted features from formula sequences and integrated them with contextual representations, highlighting the importance of text–math fusion.

The *NTCIR-12 MathIR Task* extended this setup to Wikipedia and arXiv corpora, incorporating both keyword and formula queries [8]. The ICST system introduced a hybrid indexing model that leveraged semantic operator trees and RankBoost for ranking, achieving a Precision@5 of 0.4733 on Wikipedia queries [9]. These findings illustrated the advantage of structural weighting combined with learning-to-rank methods for large-scale MIR.

Community-driven benchmarks have further shaped the field. The *ARQMath 2020 Lab* introduced mathematical question answering and formula retrieval using Math Stack Exchange data [10]. Task 1 (Answer Retrieval) covered 77 topics, while Task 2 (Formula Retrieval) focused on 45 topics, evaluated with *nDCG* and MAP. Systems such as DPRL3 and zbMATH integrated textual features with formula similarity, but reported peak *nDCG* values of only 0.042, reflecting the difficulty of retrieval in noisy, user-generated CQA content [11].

Algorithmic innovations have also shaped MIR. The Tangent search engine introduced Maximum Subtree Similarity (MSS), where formulae were represented via symbol pair indexing and compact structural encodings [12]. Evaluated on the NTCIR-11 benchmark, Tangent achieved a p@5 of approximately 92%, demonstrating that subtree-based retrieval could balance efficiency with high retrieval quality. Techniques such as MSS subsequently influenced hybrid retrieval architectures by showing that compact structural representations scale well in practice.

The application of neural architectures has introduced new perspectives into MIR. Structure-aware deep models have been developed to capture the syntactic properties of formulae, while semantic em-

beddings allowed for generalization across notational variants. Recursive and graph-based embeddings demonstrated the potential of bridging symbolic structure with semantic equivalence [13, 14]. At the same time, contextual embeddings from transformers enabled dense representations of surrounding text, aligning formulae with their semantic usage [15, 16].

Parallel work in cross-lingual information retrieval (CLIR) provides relevant foundations for extending MIR to multilingual settings. Dictionary-based alignment methods and statistical translation models established early baselines, while vector space and positional language models refined ranking strategies [17, 18]. More recently, multilingual transformers such as mBERT and XLM-R demonstrated strong cross-lingual transfer without requiring parallel corpora [19, 20]. Region-specific pretraining has further improved retrieval for Indian languages, with IndicBERT and MuRIL handling code-mixing, transliteration, and other linguistic phenomena common in South Asian contexts [21, 22].

Our work builds upon these strands of research by combining multilingual semantic embeddings with symbolic reasoning for mathematical expressions. Specifically, we introduce a hybrid retrieval model that leverages sentence-transformer representations for cross-lingual alignment, SymPy-based similarity for formula comparison, and FAISS indexing for scalable retrieval. By integrating symbolic and neural methods within a unified framework, our approach aims to address the limitations of purely monolingual or symbolic systems and establish a more robust baseline for cross-lingual mathematical information retrieval.

3. Dataset

The dataset used in this study is released as part of the FIRE-2025 CLMIR shared task. It is derived from the Math StackExchange corpus used in ARQMath-1 and has been adapted to support cross-lingual retrieval between English and Hindi. The training corpus, provided in `Train_set.csv`, consists of 39,862 mathematical questions and answers written primarily in Hindi, enriched with LaTeX-formatted mathematical expressions. Each entry is organized into four fields: a unique identifier, a Hindi title describing the problem, a body containing explanatory text and embedded formulas, and a set of topical tags covering domains such as probability, algebra, group theory, graph theory, etc. The titles and tags suggest that many questions are translations or adaptations of English mathematical queries, offering a localized benchmark for cross-lingual evaluation.

The body of posts typically contains descriptive text in Hindi, often mixed with English terminology, along with LaTeX-formatted mathematical content. Examples range from probability calculations and group-theoretic questions to systems of congruences and functional equations, making the dataset suitable for retrieval tasks that integrate both textual and symbolic reasoning. Tags further categorize posts into mathematical subfields, facilitating filtering and thematic analysis. Table 1 summarizes structure of the training dataset.

The official test set, `Test_Data.csv`, comprises 50 English queries designed to retrieve relevant Hindi posts from the training collection. Each query consists of a unique identifier, a LaTeX-formatted mathematical expression, and a short English description indicating its mathematical domain. Queries span a diverse range of topics, including geometry (e.g., the Pythagorean theorem), calculus (e.g., definite integrals and derivatives), linear algebra (e.g., eigenvalues and dot products), statistics (e.g., variance), partial differential equations (e.g., Laplace and heat equations), etc. as represented in Table 2.

The combination of Hindi documents and English queries highlights the challenges of cross-lingual and math-heavy retrieval, motivating approaches that effectively integrate multilingual semantic representations with symbolic formula understanding.

4. Methodology

We address the problem of CLMIR, where an English query must retrieve Hindi mathematical documents containing both natural language and LaTeX-based formulae. Formally, each query is repre-

Table 1
Sample representation of Train_set.csv for CLMIR 2025

Id	Title	Body (Summary)	Tags
1	पिछले संख्याओं के आधार पर अगले यादृच्छिक संख्या की संभावना	Calculates probabilities for the next digit (0-9) in a random sequence, e.g., 0: 10.125%, 1: 9.25%, given prior digits: 0, 2, 3, 4, 6, 4, 4, 9, 1, 3, 5, 5, 8, 7, 2.	संभावना
2	मान लीजिए g गुणक समूह को दर्शाता है $\{-1, 1\}$	Defines group $g = \{-1, 1\}$ and set $s = \{z \in \mathbb{C} : \lvert z\rvert = 1\}$, with g acting on s via complex multiplication. Asks for the cardinality of i's orbit: a) 1, b) 2, c) 5, d) ∞.	समूह-सिद्धांत
		.....	
9997	टोपोलॉजी की परिभाषा में खुले सेट के बारे में थोड़ी समस्या	Questions if a finite isolated point is an open set in a topology defined by a collection $u = \{u_\alpha\}$ of subsets of X. Suggests it might be a closed set due to unclear open set definition.	सामान्य स्तरीय विज्ञान
9998	इस तरह का सबग्राफ कैसे कहा जाता है?	Describes a directed graph G with a strongly connected subgraph s where all paths from G's nodes lead to s and remain there. Asks for the name of such a subgraph acting as a sink.	ग्राफ सिद्धांत
10000	रैखिक बधाई की एक प्रणाली को हल करें	Presents a system of linear congruences: $a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \equiv b_1 \pmod p$, $\ldots$, $a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n \equiv b_n \pmod p$. Questions the validity of row operations.	रैखिक-बीजगणित, संख्या-सिद्धांत, मॉड्यूलर-एरिथमेटिक

Table 2
Selected entries from the `Test_Data.csv` dataset, illustrating English queries for CLMIR 2025.

Query ID	Query	Context
q_1	$a^2 + b^2 = c^2$	Pythagorean theorem in geometry
q_5	$\int_a^b f(x)\,dx$	Definite integral in calculus
q_13	$A \cdot x = \lambda \cdot x$	Eigenvalue equation in linear algebra
q_14	$\sum_{i=1}^{n}(x_i - \bar{x})^2$	Variance in statistics

sented as a pair

$$q = (q_t, q_m),$$

where $q_t \in \Sigma_{\text{en}}$ denotes the textual component and q_m is a (possibly empty) LaTeX expression. Each document is similarly represented as

$$d = (d_t, d_m),$$

where $d_t \in \Sigma_{\text{hi}}$ corresponds to the textual part in Devanagari script and d_m is a set or sequence of LaTeX expressions embedded in the document body.

The retrieval objective is to compute a hybrid similarity score

$$S(q, d) \;=\; \alpha\, s_{\text{text}}(q_t, \tau(d_t)) \;+\; (1 - \alpha)\, s_{\text{math}}(q_m, d_m),$$

where τ is a transliteration function mapping Devanagari text into Latin script, s_{text} measures semantic similarity between query and document text, s_{math} measures symbolic similarity between the corre-

sponding mathematical expressions, and $\alpha \in (0, 1)$ balances the two components. For each query, the system first retrieves a candidate set of documents based on text similarity, and then re-ranks them using the hybrid score to produce the final top-R results.

4.1. Preprocessing

The preprocessing stage begins by separating textual and mathematical components from both queries and documents. LaTeX expressions are extracted using common inline and display delimiters. These extracted spans form the mathematical parts (q_m, d_m), while the surrounding content constitutes the textual parts (q_t, d_t). Any malformed or unparsable fragments are discarded.

Textual normalization differs slightly for English and Hindi. English text is lowercased, stripped of punctuation, and cleaned for consistent whitespace. Hindi text is first transliterated into Latin script using the ITRANS scheme (via the `indic-transliteration` library). This ensures compatibility with English-trained embedding models while preserving semantic meaning (e.g., "समाकलन" → "samakalan"). Further cleaning removes diacritics and harmonizes character variants. The text is then tokenized with the model-specific tokenizer, yielding encoder-ready input.

To maintain input quality, documents with fewer than ten tokens or those consisting entirely of mathematical expressions are excluded. Additionally, input length is capped at 512 tokens to comply with transformer encoder limits.

4.2. Embedding Generation and Candidate Retrieval

Once preprocessed, the textual components of queries and documents are mapped into dense vector embeddings using Sentence Transformer models. Each encoder outputs a normalized representation, and relevance is measured through cosine similarity:

$$S_t(q, d) = \cos(v_q^t, v_d^t).$$

This formulation ensures that semantically aligned query-document pairs yield higher scores, while irrelevant ones are pushed toward lower similarity.

Efficient large-scale retrieval is achieved with FAISS indexing. The embedding space is partitioned into clusters using k-means, and only the most relevant clusters are searched at query time. This approach balances speed and accuracy, producing the top $K = 500$ candidate documents for each query. These candidates are then subject to hybrid re-ranking.

4.3. Symbolic (Math) Similarity

In addition to textual matching, mathematical content is treated as a first-class signal of relevance.

Run 1 employs a lightweight strategy: expressions are tokenized into operators, variables, and symbols, and similarity is measured by token overlap,

$$S_m(q, d) = \frac{|T(q_m) \cap T(d_m)|}{|T(q_m)|},$$

where $T(\cdot)$ denotes the extracted set of math tokens. This method prioritizes efficiency and rewards exact or near-exact matches.

Run 2 applies a more expressive, structure-aware approach. Here, mathematical expressions are parsed into operator-operand trees. Similarity is then computed by comparing overlapping nodes in these trees, allowing the system to capture correspondences between expressions that differ syntactically but remain structurally or semantically related.

4.4. Run Configurations

Although the overall retrieval pipeline is shared across both runs, they diverge significantly in three aspects: the choice of embedding models, the configuration of FAISS indexing, and the balance between textual and mathematical similarity when computing the final hybrid score. These differences reflect distinct design priorities, with Run 1 focusing on efficiency and scalability, while Run 2 aims for higher retrieval accuracy and stronger cross-lingual robustness.

Run 1 (MiniLM-based): This configuration adopts `multi-qa-MiniLM-L6-cos-v1`, a compact Sentence Transformer model that produces 384-dimensional embeddings. The main advantage of MiniLM lies in its efficiency. The smaller embedding size reduces memory usage, allowing the system to index a large number of documents without excessive storage overhead. Moreover, its lightweight architecture accelerates both the encoding and retrieval stages, making it suitable for scenarios where computational resources are limited or where rapid response times are critical.

Candidate retrieval in Run 1 is implemented using FAISS with a fixed configuration, where the embedding space is partitioned into a predetermined number of clusters ($n_{\text{list}} = 100$) and a fixed number of probes ($n_{\text{probe}} = 10$) are searched at query time. This static setting ensures consistent query latency across the entire collection. While such an approach may occasionally sacrifice recall—since only a limited subset of clusters is explored—it guarantees predictability and efficiency, both of which are desirable for real-time or large-scale applications.

On the mathematical side, Run 1 employs the token-overlap method already described in the previous subsection. Since this strategy does not capture deeper equivalences between expressions, the hybrid scoring function deliberately places greater emphasis on the textual channel. The final similarity score is computed with a 0.7 weight on text and 0.3 weight on math:

$$S(q, d) = 0.7 \cdot S_t(q, d) + 0.3 \cdot S_m(q, d).$$

This choice reflects the design philosophy of Run 1: textual similarity is treated as the primary signal of relevance, while mathematical similarity plays a secondary, supportive role. In practice, this setup enables rapid large-scale retrieval while still leveraging mathematical information for disambiguation in cases where textual evidence alone is insufficient.

Run 2 (MPNet-based): The second configuration takes a different stance, prioritizing robustness and retrieval quality over raw speed. For the textual encoder, we employ `all-mpnet-base-v2`, which generates 768-dimensional embeddings. MPNet offers stronger representational capacity compared to MiniLM, making it more effective at capturing fine-grained semantic relationships. More importantly, it has demonstrated better performance in multilingual and cross-lingual settings, which is particularly important here, since English queries must retrieve documents written in Hindi. The trade-off, however, is higher computational cost and larger memory requirements due to the increased dimensionality.

Candidate generation in Run 2 also differs from Run 1 in its use of an adaptive FAISS indexing strategy. Rather than fixing the number of clusters and probes in advance, the index parameters are scaled with the size of the document collection. Specifically, the number of clusters increases with corpus size, ensuring finer partitioning of larger collections, while the number of probes is also adjusted dynamically to improve recall. This adaptive mechanism enhances robustness: when applied to small datasets, it avoids unnecessary overhead, but on larger datasets it improves retrieval coverage, reducing the risk of missing relevant documents that might fall outside the probed clusters.

For the mathematical similarity component, Run 2 integrates the more expressive tree-based structural approach introduced earlier. Since this strategy is capable of capturing structural and semantic correspondences between expressions, the hybrid scoring function gives greater relative weight to the mathematical channel compared to Run 1. The final score is defined as:

$$S(q, d) = 0.6 \cdot S_t(q, d) + 0.4 \cdot S_m(q, d).$$

Here, textual similarity remains slightly dominant, but mathematical similarity is treated as a more substantial factor, reflecting the configuration's emphasis on the dual importance of both language and symbolic reasoning in cross-lingual mathematical retrieval.

4.5. Hybrid Re-ranking

For both runs, the system first retrieves 500 candidates using text similarity alone. These candidates are then re-ranked using the hybrid scoring function $S(q, d)$. Finally, the top $R = 50$ ranked documents are returned. In the event of tied scores, preference is given to the document with higher text similarity.

Algorithm 1 Hybrid Retrieval Model based on MiniLM for CLMIR (Run 1)

1: **Input:** English query $q = (q_t, q_m)$, Hindi document collection $\mathscr{D} = \{d_1, d_2, \ldots, d_N\}$
2: **Textual Preprocessing:** Normalize text, remove stopwords, perform stemming
3: **Mathematical Preprocessing:** Extract LaTeX formulae, tokenize into operators, variables, and structures
4: Encode textual part of query and documents using MiniLM:

$$v_q^t = f_{\text{MiniLM}}(q_t), \quad v_d^t = f_{\text{MiniLM}}(d_t)$$

5: Compute textual similarity:

$$S_t(q, d) = \cos(v_q^t, v_d^t)$$

6: Encode mathematical part with regex-based token matching:

$$S_m(q, d) = \frac{|T(q_m) \cap T(d_m)|}{|T(q_m)|}$$

where $T(\cdot)$ denotes extracted math tokens
7: Combine similarities with linear interpolation:

$$S(q, d) = \lambda \cdot S_t(q, d) + (1 - \lambda) \cdot S_m(q, d)$$

8: Rank documents $\mathscr{D}$ in descending order of $S(q, d)$
9: **Output:** Ranked list of documents $\{d_{(1)}, d_{(2)}, \ldots, d_{(N)}\}$

Algorithm 2 Hybrid Retrieval Model based on MPNet for CLMIR (Run 2)

1: **Input:** English query $q = (q_t, q_m)$, Hindi document collection $\mathscr{D} = \{d_1, d_2, \ldots, d_N\}$
2: **Textual Preprocessing:** Normalize text, remove stopwords, perform stemming
3: **Mathematical Preprocessing:** Extract LaTeX formulae, represent expressions as operator trees
4: Encode textual part of query and documents using MPNet:

$$v_q^t = f_{\text{MPNet}}(q_t), \quad v_d^t = f_{\text{MPNet}}(d_t)$$

5: Compute textual similarity:

$$S_t(q, d) = \cos(v_q^t, v_d^t)$$

6: Encode mathematical part via tree-based structural similarity:

$$S_m(q, d) = \frac{|N(q_m) \cap N(d_m)|}{|N(q_m)|}$$

where $N(\cdot)$ denotes set of operator–operand nodes in expression trees
7: Combine similarities with weighted sum:

$$S(q, d) = \mu \cdot S_t(q, d) + (1 - \mu) \cdot S_m(q, d)$$

8: Rank documents $\mathscr{D}$ in descending order of $S(q, d)$
9: **Output:** Ranked list of documents $\{d_{(1)}, d_{(2)}, \ldots, d_{(N)}\}$

5. Results

Table 3 summarizes the official CLMIR 2025 Shared Task scores for all participating teams. The evaluation used three standard metrics: Precision@10 (P@10), Mean Average Precision (MAP) and normalized Discounted Cumulative Gain (nDCG). IReL submitted two runs whose scores exhibit a striking contrast; below we describe these results in detail, quantify the improvements, and analyse likely causes and implications.

Table 3
Performance metrics for all teams in the CLMIR 2025 Shared Task.

Team	Run	P@10	MAP	nDCG
Archisha Dhyani	Run 1	0.028	0.0712	0.1002
	Run 2	0.044	0.0941	0.1224
	Run 3	0.050	0.1034	0.1457
DUCS_CLMIR	Run 1	0.006	0.0145	0.0563
	Run 2	0.008	0.0159	0.0567
	Run 3	0.010	0.0153	0.0517
IReL	**Run 1**	**0.000**	**0.0064**	**0.0352**
	Run 2	**0.122**	**0.1650**	**0.3063**
NLP Fusion	Run 1	0.048	0.0972	0.1521
	Run 2	0.046	0.0794	0.1966
	Run 3	0.068	0.1090	0.2523
	Run 4	0.118	0.1490	0.2898
Retriever	Run 1	0.000	0.0016	0.0089
	Run 2	0.114	0.1755	0.3031
	Run 3	0.128	0.2143	0.3264
	Run 4	0.072	0.1003	0.2227
Tends	Run 1	0.080	0.1484	0.2202
Organizer	Run 1	0.085	0.1380	0.2450

IReL's Run 2 shows a clear and substantial improvement over Run 1 across all reported metrics. Run 1 recorded P@10 = 0.000, MAP = 0.0064 and nDCG = 0.0352, whereas Run 2 achieved P@10 = 0.122, MAP = 0.165 and nDCG = 0.3063. The P@10 jump from 0.000 to 0.122 is particularly important for user-facing retrieval, as it reflects many more relevant documents appearing in the top-10 returned items.

When compared against other teams, IReL's Run 2 is competitive with the leading submissions. Its P@10 of 0.122 surpasses Archisha Dhyani's runs (0.028-0.050), all DUCS_CLMIR runs (0.006-0.010), and Tends (0.080); it also outperforms most NLP Fusion runs (0.046-0.068) and is comparable to NLP Fusion's strongest run (0.118) and Retriever's Run 2 (0.114). In MAP, Run 2 (0.165) ranks above Archisha Dhyani (0.0712-0.1034) and DUCS_CLMIR (0.0145-0.0159), and is competitive with the stronger NLP Fusion and Retriever runs (whose MAPs reach up to 0.2143). For nDCG, Run 2 (0.3063) is among the top performers: it substantially outperforms most teams and is only slightly below Retriever's best submission (Run 3, nDCG = 0.3264). Taken together, these comparisons indicate that Run 2 attained high ranking quality relative to the shared task field, especially in nDCG which measures the graded quality of top-ranked results.

A closer inspection of Run 1 shows several failure modes that explain its very low scores. Similarity values produced by Run 1 span a wide but misleading range: some queries (e.g., q_1, q_2, q_5, q_46, q_48, q_50) yielded relatively high similarity scores (reported values between approximately 0.2275 and 0.8500), yet these high scores did not translate into correct, diverse top-ranked documents. Instead, Run 1 frequently returned identical document sets for multiple distinct queries (for example, the exact same top candidates for q_1, q_2, q_5, q_46, q_48 and q_50), which strongly suggests insufficient

query-specific discrimination. One striking artifact is document `15243`, which attains a very high similarity (0.85000014) across many queries; this behaviour is consistent with an indexing or retrieval bias. These issues explain why Run 1's precision and ranking measures are near the bottom of the leaderboard despite sporadically high pairwise similarity scores.

By contrast, Run 2 shows both empirically and qualitatively improved behaviour. The range and distribution of similarity scores for Run 2 are more informative and query-sensitive (for instance, reported similarity ranges include 0.165-0.6017 for `q_1` and 0.4587-0.5921 for `q_2`), indicating better differentiation among candidate documents. Document-overlap analysis between the two runs confirms this: in many cases Run 2 returns different and more relevant documents than Run 1 (e.g., `q_1` retrieves `4446`, `33248`, ... under Run 2 instead of `15243`, `39015`, ... under Run 1). The improved P@10 (0.122) and nDCG (0.3063) quantify these gains at the top ranks, while the MAP increase demonstrates better overall ranking across the full list. That said, Run 2 is not uniformly strong for every query: an interesting anomaly is `q_3`, for which Run 2 reports relatively low similarity values (approximately 0.2432-0.3309). This suggests that `q_3` contains particularly challenging content (for example, unusually complex formulae or mixed-language phenomena) that evade the model's current alignment, and it highlights remaining per-query weaknesses even in the stronger run.

We also categorized query-level performance by topic domains such as calculus, algebra, and geometry. Preliminary inspection revealed stronger performance for algebraic and linear equation queries, where symbolic structures align closely, compared to geometric or descriptive text-heavy queries, which rely more on semantic cues. This suggests that symbolic similarity contributes disproportionately in formula-centric categories. Conversely, performance degradation for mixed or linguistically rich queries highlights the need for adaptive weighting or task-specific fine-tuning.

In summary, IReL's submissions demonstrate a dramatic within-team improvement from Run 1 to Run 2. Run 2 attains competitive performance among the shared task participants, particularly in nDCG, and achieves substantial absolute gains in MAP and P@10 relative to Run 1. However, the detailed per-query analysis reveals both systematic issues (indexing/retrieval bias in Run 1) and remaining edge-case failures (e.g., `q_3`) that motivate further refinement. These observations underscore that gains in cross-lingual mathematical retrieval arise from (i) improving the fidelity of math-to-math matching, (ii) ensuring indexing and candidate retrieval are not dominated by a small set of artifacts, and (iii) addressing difficult, query-specific phenomena through targeted modelling and evaluation strategies.

6. Conclusion and Future Work

The CLMIR 2025 Shared Task highlighted the challenges of retrieving Hindi mathematical documents from English queries that combine text with symbolic expressions. The IReL system explored two approaches: the first run, based on a lightweight multilingual encoder and binary math similarity, performed poorly with MAP = 0.0064, P@10 = 0.000, and nDCG = 0.0352, underscoring the limitations of such models in cross-lingual math retrieval. In contrast, the second run, which incorporated the `all-mpnet-base-v2` model, dynamic FAISS indexing, and a hybrid scoring scheme combining text and formula similarity, achieved substantially stronger results with MAP = 0.165, P@10 = 0.122, and nDCG = 0.3063. These findings emphasize the importance of balancing semantic and symbolic signals for effective retrieval. Remaining challenges include handling complex mathematical structures, mitigating transliteration noise, and optimizing the relative contribution of text and math components. While evaluated primarily on the FIRE CLMIR dataset, the proposed hybrid framework can be readily extended to other multilingual math corpora (e.g., English–Tamil or English–Bengali), given that the underlying components (transliteration, semantic encoding, and symbolic reasoning) are language-agnostic. This positions the system as a flexible foundation for broader multilingual scientific information retrieval. Future research directions involve fine-tuning transformer embeddings on math-rich multilingual corpora, developing richer formula similarity measures that integrate structural and numerical evaluations, and employing learning-to-rank strategies to adaptively weight hybrid scores. Ex-

ploring query expansion, ensemble retrieval, and external knowledge integration can further improve robustness and scalability in multilingual mathematical information retrieval.

Declaration on Generative AI

During the preparation of this work, the authors used ChatGPT, Grammarly in order to: Grammar and spelling check, Paraphrase and reword. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] A. Youssef, Roles of math search in mathematics, in: Proceedings of the Symposium on Computer Algebra and Scientific Computing, 2005.

[2] R. Zanibbi, D. Blostein, Recognition and retrieval of mathematical expressions, in: International Conference on Document Analysis and Recognition, IEEE, 2012, pp. 145–154.

[3] Z. Mihalinec, P. Sojka, Mathsearch: A search engine for mathematical content, in: Proceedings of the International Conference on MathML and Technologies for Math on the Web, 2002.

[4] P. Sojka, M. Liska, Indexing and searching mathematics in digital libraries, Mathematics in Computer Science 5 (2011) 227–241.

[5] A. Youssef, Methods of relevance ranking and hit-content generation in math search, in: Proceedings of CICM, 2006.

[6] D.-D. Nguyen, H.-H. Nguyen, et al., An approach to searching mathematical content in vietnamese, in: Asian Conference on Intelligent Information and Database Systems, Springer, 2012, pp. 58–67.

[7] A. Aizawa, M. Kohlhase, I. Ounis, et al., Ntcir-11 math-2 task overview, in: Proceedings of the 11th NTCIR Conference on Evaluation of Information Access Technologies, National Institute of Informatics, 2014, pp. 88–98.

[8] R. Zanibbi, A. Aizawa, I. Ounis, et al., Ntcir-12 mathir task overview, in: Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies, 2016, pp. 299–308.

[9] L. Gao, K. Yuan, Y. Wang, Z. Jiang, Z. Tang, The math retrieval system of icst for ntcir-12 mathir task, in: Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies, National Institute of Informatics, 2016, pp. 318–325.

[10] R. Zanibbi, A. Aizawa, B. Mansouri, I. Ounis, M. Schubotz, H. Stange, A. Youssef, Overview of the arqmath 2020 competition on answer retrieval for questions on math, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction (CLEF 2020), Springer, 2020, pp. 169–193. doi:10.1007/978-3-030-58219-7_12.

[11] B. Mansouri, R. Zanibbi, A. Aizawa, I. Ounis, M. Schubotz, H. Stange, A. Youssef, Overview of arqmath task 1 and 2: Answer retrieval and formula retrieval, in: Working Notes of CLEF 2020, 2020.

[12] R. Zanibbi, K. Davila, M. Schubotz, et al., Tangent-cft: An improved search engine for mathematical formulae, in: Proceedings of the 39th International ACM SIGIR Conference, 2016, pp. 1165–1168.

[13] T. Schellenberg, R. Zanibbi, Tangent-l: An embedding model for mathematical formulas, in: Proceedings of the 44th International ACM SIGIR Conference, 2021, pp. 1579–1583.

[14] X. Zheng, C. Lin, E. Tang, Mathematical formula retrieval with structure-aware deep neural networks, in: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics, 2021, pp. 2549–2560.

[15] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, in: Proceedings of EMNLP, 2019.

[16] A. Conneau, et al., Cross-lingual language model pretraining, in: Advances in Neural Information Processing Systems, 2019.

[17] G. Salton, A. Wong, C.-S. Yang, A vector space model for automatic indexing, Communications of the ACM 18 (1975) 613–620.
[18] Y. Lv, C. Zhai, Positional language models for information retrieval, in: Proceedings of the 34th International ACM SIGIR Conference, 2011, pp. 299–306.
[19] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of NAACL-HLT, 2019.
[20] A. Conneau, et al., Unsupervised cross-lingual representation learning at scale, in: Proceedings of ACL, 2020.
[21] D. Kakwani, et al., Indicnlpsuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for indian languages, in: Proceedings of EMNLP, 2020.
[22] S. Khanuja, et al., Muril: Multilingual representations for indian languages, in: Findings of EMNLP, 2021.