MaRF: Leveraging Representation-Level Fusion of Formula Semantics for Mathematical Information Retrieval

Suyuan Wang* **Hongbo Zheng*** **Nickvash Kani**
University of Illinois at Urbana-Champaign
{suyuan2, hongboz2, kani}@illinois.edu

Abstract

Mathematical information retrieval (MIR) depends on jointly modeling natural-language context and mathematical expressions. While BERT-based dense retrievers are effective, they often dilute mathematical semantics because textual content dominates most training data and mathematical formulas differ fundamentally from natural language in structure and composition. Consequently, these models rely heavily on surrounding text, which reduces robustness in math-intensive scenarios with limited textual description. We propose MaRF, a dual-encoder representation-level fusion framework for MIR that explicitly integrates formula semantics into context-aware dense retrieval. By combining contextual and formula-specific representations, MaRF captures complementary information from both textual and symbolic views. Experiments on the ARQMath-3 benchmark demonstrate that MaRF substantially improves retrieval performance and robustness, outperforming strong baselines across MIR tasks. The source code and datasets are available at https://github.com/MLPgroup/MaRF.

1 Introduction

As the volume of scientific literature continues to grow, scholarly documents increasingly contain rich and complex mathematical content. This trend has made the representation, understanding, and retrieval of mathematical information an increasingly important problem, drawing sustained attention from the mathematical information retrieval (MIR) community (Zanibbi and Blostein, 2012; Zanibbi et al., 2025). MIR aims to retrieve relevant information from documents containing mathematical content, such as formulas, derivations, and their surrounding textual explanations.

Compared with standard natural language retrieval, MIR presents several distinct challenges. First, mathematical formulas form a highly structured symbolic language whose syntax differs fundamentally from natural language. Expressions may include hierarchical and multi-line constructs, such as fractions, superscripts, and subscripts, that are difficult to represent using standard text-based encoders. Second, mathematical expressions are often highly variable: a single formula can admit many mathematically equivalent rewrites under transformation rules, while its surface form and visual structure may change substantially. For example, $\sec(x)/\sin(x)$ and $\tan(x) + 1/\tan(x)$ are mathematically equivalent despite having very different syntactic forms. This variability limits the effectiveness of structure-search and exact or near-exact matching approaches (Pavan Kumar et al., 2012), which often struggle to capture equivalence under transformation. Third, the same mathematical expression may carry different meanings across domains (Hamel et al., 2022), making the surrounding textual context essential for resolving ambiguity and recovering document-level semantics.

A widely adopted approach to MIR is dense retrieval (Karpukhin et al., 2020; Zhong et al., 2022a), which encodes queries and documents into dense vectors using Transformer-based encoders (Vaswani et al., 2017) and retrieves candidates based on embedding similarity. Compared with structure-based retrieval methods (Kane et al., 2022; Zhong et al., 2022a,b; Geletka et al., 2022; Mansouri et al., 2022a,b), dense retrievers can better exploit contextual descriptions and capture higher-level semantic relationships, leading to improved robustness in many settings. However, existing dense retrievers still face an important limitation in MIR. Mathematical documents are inherently heterogeneous: they interleave natural-language explanations with symbolic expressions, while textual tokens often dominate the overall input. Because mathematical formulas differ substantially from natural language in syntax and com-

*Equal Contribution.

Findings of the Association for Computational Linguistics: ACL 2026, pages 25570–25585

position, their semantics can be underrepresented during encoder training and attention computation. As a result, dense retrievers tend to rely heavily on surrounding text, while their representations of mathematical content remain comparatively weak. This limitation becomes especially pronounced in math-intensive scenarios where textual descriptions are sparse, such as long derivations in textbooks, lecture notes, and research papers. As we show in Section 4.3, conventional dense retrievers degrade substantially on math-dominated retrieval cases, where mathematical tokens constitute the majority of the content. These observations highlight the need for retrieval models that explicitly strengthen the representation of formula semantics.

Recent work (Gangwar and Kani, 2022; Zheng et al., 2025) has shown that contrastive learning over formulas can effectively capture mathematical similarity and transformation-aware semantics. Motivated by this line of research, we propose **MaRF**, a dual-encoder representation-level fusion framework for MIR. MaRF combines a context-aware dense retriever with a lightweight formula encoder specialized for mathematical semantics, and integrates the two through a fusion module that injects formula-level information into contextual representations. In this way, MaRF preserves the strengths of contextual retrieval while explicitly enhancing the modeling of symbolic mathematical structure. We evaluate MaRF on the ARQMath-3 benchmark (Mansouri et al., 2022a) and further examine its generality on NTCIR-12 (Zanibbi et al., 2016). In addition, we study MaRF with both a BERT-based backbone and an LLM-style backbone. Across these settings, MaRF consistently improves retrieval effectiveness and robustness, particularly in math-intensive scenarios where standard dense retrievers are more likely to fail.

The contributions of this work are summarized as follows:

❶ We propose MaRF, a representation-level fusion framework for MIR that explicitly integrates formula semantics into context-aware dense retrieval, improving robustness in math-intensive retrieval settings.

❷ We investigate the design of the formula modeling component, including passage-level and token-level scoring strategies, and analyze their effects on learning mathematical semantic similarity.

❸ We show that MaRF consistently improves retrieval performance across MIR benchmarks and backbone choices, demonstrating the effectiveness of combining contextual and formula-specific representations.

2 Related Work

Structure-Based Retrieval. Structure-based retrieval has long been a central paradigm in mathematical information retrieval (MIR) (Miller and Youssef, 2003), with methods explicitly modeling the structural organization of mathematical expressions. Representative systems such as MathDowsers (Fraser et al., 2018; Ng, 2021; Kane et al., 2022) and Tangent-S (Mansouri et al., 2022a) extract features from structured representations, including Symbol Layout Trees (Schellenberg et al., 2012; Zanibbi and Blostein, 2012), operator trees, and MathML, enabling retrieval based on hierarchical and spatial relationships between symbols. Other approaches, such as Approach0 (Zhong and Zanibbi, 2019; Zhong et al., 2020) and MSM/BM25 (Novotnỳ et al., 2021; Rohatgi et al., 2021), integrate structural signals into classical information retrieval frameworks using BM25 (Robertson and Walker, 1994) or TF-IDF (Sparck Jones, 1972) to support efficient search. Although these methods are effective at matching explicit structural patterns, they often struggle to capture semantic equivalence under algebraic transformation or to make use of broader contextual information.

Dense Retrieval for MIR. More recently, data-driven dense retrieval methods have been increasingly adopted in MIR, ranging from earlier embedding-based approaches (Gao et al., 2017; Mansouri et al., 2019) to Transformer-based models (Peng et al., 2021; Reusch et al., 2021a). Compared with structure-based retrieval, dense retrievers can better exploit surrounding textual context and learn higher-level semantic representations, which improves robustness in settings where exact structural matching is insufficient. Representative models include QA-Sim (Mansouri et al., 2021a) and the ALBERT-based model (Reusch et al., 2021b), both of which adopt cross-encoder architectures and therefore incur relatively high inference cost. To improve scalability, later work explored bi-encoder frameworks, including CompuBERT (Novotnỳ et al., 2021) and FormulaEmb (Dadure et al., 2021). In addition, In+Out (Amador and Zanibbi, 2025) adopts a bi-encoder design to jointly learn visual and semantic embeddings for mathematical formulas. Despite

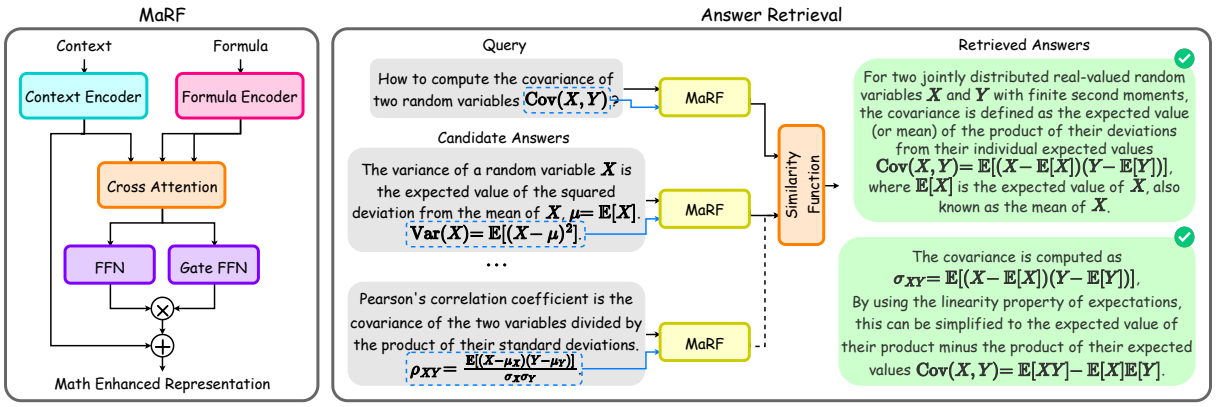

Figure 1: Outline of MaRF framework and answer retrieval. For each query or answer post, mathematical formulas are extracted and encoded by a formula encoder, while context is encoded by BERT encoder. A cross-attention based fusion layer selectively inject mathematical information from formula representations into the context representation, producing a math enhanced representation for retrieval.

their strong empirical performance, most dense MIR models rely on a single encoder to represent heterogeneous inputs containing both natural language and mathematical expressions, which can make fine-grained formula semantics difficult to preserve. This limitation motivates our use of a representation-level fusion framework that explicitly strengthens mathematical semantic modeling within dense retrieval.

Math-Aware Pretraining and Formula Modeling. Several prior studies have shown that mathematical formulas benefit from dedicated modeling strategies beyond standard text encoders. For example, MathBERT (Peng et al., 2021) adapts BERT pretraining to mathematical content, while other work has explored contrastive learning and semantic representation learning over formulas (Gangwar and Kani, 2022; Zheng et al., 2025). These studies suggest that mathematical expressions contain distinct structural and semantic patterns that are not fully captured when formulas are treated as plain text. Our work is complementary to this line of research: we use a specialized formula encoder to provide mathematical semantics that can be fused with context-aware document representations.

LLM-Based MIR. Recent work has also begun to explore large language models (LLMs) for MIR. Earlier approaches typically used generative LLMs as rerankers on top of an initial retrieval stage, since autoregressive models are not naturally designed to produce dense retrieval embeddings (Satpute et al., 2024; Mansouri and Maarefdoust, 2024). With the emergence of embedding-oriented LLMs, it has become increasingly practical to apply stronger general-purpose embedding backbones to MIR more directly. This development raises an important question: whether explicit formula-aware fusion remains beneficial when the retrieval backbone itself is substantially stronger. Our work addresses this question by evaluating MaRF not only with a BERT-based encoder, but also with a modern embedding model.

3 Methodology

Figure 1 illustrates **Ma**thematical **R**epresentation-level **F**usion (**MaRF**), a dual-encoder framework for mathematical information retrieval. MaRF consists of three components: a context encoder, a formula encoder, and a representation-level fusion module. The context encoder models document-level semantics from mixed textual and mathematical content, while the formula encoder is designed to capture mathematical relationships and transformation-aware similarities between formulas. The fusion module integrates these complementary representations by injecting formula-aware information into the contextual embedding space. In this way, MaRF preserves the strengths of context-aware dense retrieval while explicitly enhancing the modeling of mathematical semantics.

3.1 Context Encoder

The context encoder follows the standard dense retrieval pipeline in prior MIR work (Reusch et al., 2022; Zhong et al., 2022a). Given a query and a candidate document, it produces contextualized token-level embeddings for similarity computation.

In our experiments, we instantiate the context encoder with either a BERT-based backbone (Devlin et al., 2019) or Qwen3-Embedding-0.6B (Zhang et al., 2025), a strong embedding model for retrieval. In both cases, queries and documents are encoded using shared parameters.

Formally, let $\boldsymbol{x}_q \in \mathbb{R}^{N_q}$ and $\boldsymbol{x}_d \in \mathbb{R}^{N_d}$ denote the tokenized query and document sequences, respectively, where N_q and N_d are the corresponding sequence lengths. Let f_θ denote the context encoder parameterized by θ. The query and document embeddings are computed as

$$\boldsymbol{E}_q = f_\theta(\boldsymbol{x}_q), \quad \boldsymbol{E}_d = f_\theta(\boldsymbol{x}_d), \tag{1}$$

where $\boldsymbol{E}_q \in \mathbb{R}^{N_q \times d}$, $\boldsymbol{E}_d \in \mathbb{R}^{N_d \times d}$, and d is the hidden dimension.

Instead of representing an entire sequence using only a special sequence-level token (e.g., `[CLS]` in BERT or `[EOS]` in Qwen3-Embedding-0.6B) or a pooled embedding, we adopt the MaxSim scoring function (Khattab and Zaharia, 2020), which preserves token-level representations and enables fine-grained matching between query and document tokens.

$$\mathrm{MaxSim}(\boldsymbol{E}_q, \boldsymbol{E}_d) = \sum_{i=1}^{N_q} \max_{1 \le j \le N_d} \boldsymbol{E}_{q_i} \boldsymbol{E}_{d_j}^{\intercal}. \tag{2}$$

MaxSim computes pairwise token similarities between the query and document, selects the most relevant document token $\boldsymbol{E}_{d_j}$ for each query token $\boldsymbol{E}_{q_i}$, and sums these maximum similarities to obtain the final retrieval score.

The context encoder is trained using a contrastive objective. Given a query embedding $\boldsymbol{E}_q$, a positive document embedding $\boldsymbol{E}_d^+$, and a set of negative document embeddings $\{\boldsymbol{E}_{d_i}^-\}_{i=1}^N$, we optimize

$$\mathcal{L}_{\text{ctx}} = \mathbb{E}\left[-\ln \frac{S(\boldsymbol{E}_q, \boldsymbol{E}_d^+)}{S(\boldsymbol{E}_q, \boldsymbol{E}_d^+) + \sum_{i=1}^N S(\boldsymbol{E}_q, \boldsymbol{E}_{d_i}^-)} \right], \tag{3}$$

where $S(\cdot, \cdot)$ denotes the similarity function, instantiated as MaxSim in our implementation. This objective makes the query representation to be closer to relevant documents than to irrelevant ones.

3.2 Formula Encoder

While the context encoder captures document-level semantics from mixed content, it does not explicitly specialize in modeling mathematical equivalence or transformation-aware similarity between formulas. To address this limitation, we introduce a dedicated formula encoder based on a Transformer encoder architecture (Grattafiori et al., 2024). The formula encoder is trained to model semantic relationships between mathematical expressions appearing in queries and documents.

Similar to the context encoder, the formula encoder is trained with a contrastive objective over positive and negative formula pairs. To study how mathematical semantics are best represented, we consider two scoring strategies for formula similarity. The first is a *passage-level* strategy, which aggregates token embeddings into a single formula representation by average pooling and computes similarity using an InfoNCE-style objective (Oord et al., 2018). The second is a *token-level* strategy, which applies MaxSim to preserve fine-grained token interactions between formulas. This comparison allows us to examine whether mathematical semantic matching benefits more from global formula representations or token-level alignment. We provide a detailed empirical comparison of these two strategies in Section 4.3.

3.3 Representation-Level Fusion

To combine contextual and mathematical representations, we introduce a representation-level fusion module based on cross-attention. The key idea is to treat the contextual representation as the primary retrieval representation and enrich it with information extracted from the formula representation. This asymmetric design allows the context encoder to retain document-level semantics while selectively incorporating mathematically relevant features.

Let $\boldsymbol{E}_c \in \mathbb{R}^{N_c \times d}$ denote the contextual embedding and $\boldsymbol{E}_f \in \mathbb{R}^{N_f \times d}$ denote the formula embedding, where N_c and N_f are the context and formula sequence lengths, respectively. The fusion module is defined as

$$\boldsymbol{H} = \mathrm{MHA}(\boldsymbol{Q} = \boldsymbol{E}_c, \boldsymbol{K} = \boldsymbol{E}_f, \boldsymbol{V} = \boldsymbol{E}_f), \tag{4}$$

$$\hat{\boldsymbol{H}} = \mathrm{FFN}_G(\boldsymbol{H}) \odot \mathrm{FFN}(\boldsymbol{H}), \tag{5}$$

$$\boldsymbol{O} = \hat{\boldsymbol{H}} + \boldsymbol{E}_c, \tag{6}$$

where MHA denotes multi-head cross-attention, FFN denotes a feed-forward network, FFN_G denotes a gating network, and $\odot$ is the Hadamard (element-wise) product.

In this formulation, the context representation attends to the formula representation to retrieve mathematically relevant information. The gated feed-forward transformation then controls how much

of the retrieved formula information should be injected into the contextual embedding space. Finally, a residual connection with $\boldsymbol{E}_c$ preserves the original contextual information and stabilizes optimization. The resulting fused representation $\boldsymbol{O} \in \mathbb{R}^{N_c \times d}$ is used for retrieval.

The fusion module is also trained with a contrastive objective. Given fused query and document representations, the model is optimized to assign higher similarity to relevant pairs than to irrelevant ones. In practice, the fused representations are scored using the same MaxSim-based objective as the context encoder.

3.4 Training Strategy

Context Encoder Instantiations. We consider two context encoder instantiations in this work: a BERT-based encoder and Qwen3-Embedding-0.6B. The BERT-based encoder serves as the main dense retrieval baseline and also as the backbone of the standard MaRF setting. Qwen3-Embedding-0.6B is included to examine whether the proposed fusion framework remains effective when the backbone is replaced with a stronger modern retrieval embedding model.

Further Pretraining and Fine-Tuning of BERT. The ARQMath corpus contains roughly 1,000 mathematical tokens that do not appear in the original BERT tokenizer. Moreover, standard pretrained BERT is not optimized specifically for retrieval in mathematical domains. To better adapt it to math-rich text, we further pretrain the BERT backbone on 1.7M math-text documents using the masked language modeling (MLM) and next sentence prediction (NSP) objectives (Devlin et al., 2019). The pretraining corpus is collected primarily from Math Stack Exchange and Art of Problem Solving. All documents are converted to LaTeX, split into sentence pairs, and tokenized using the PyA0 toolkit (Zhong and Lin, 2021).

After further pretraining, we fine-tune the BERT-based context encoder on query-document triplets constructed from ARQMath. For each query post q, we treat accepted answers, as well as answers with more than five upvotes as positive samples d^+, and randomly sample answers associated with different queries as negative samples d^-.

Qwen3-Embedding Backbone. For the Qwen3-Embedding-0.6B setting, we use the pretrained model directly as a strong off-the-shelf retrieval backbone. Unlike the BERT-based encoder, we do not apply additional domain-adaptive pretraining or task-specific fine-tuning. We use the last hidden states as token-level output representations and project them to a 768-dimensional space before fusion, so that they are compatible with the fusion module. This design allows us to evaluate whether MaRF provides complementary gains beyond those already offered by a modern embedding model.

Training of the Formula Encoder. To train the formula encoder, we extract formulas from query and answer posts and construct formula-level contrastive training data. For each query post, we assume that formulas appearing in accepted answers are semantically related to formulas in the query. We therefore construct training tuples of the form $(f_q, f^+, f_1^-, f_2^-, \ldots, f_N^-)$, where f_q denotes a formula from the query post, f^+ is a formula sampled from an accepted answer, and $\{f_i^-\}_{i=1}^N$ are negative formulas sampled from irrelevant answer posts.

Training of the Fusion Layer. After training the context encoder and formula encoder, we freeze both encoders and train only the lightweight fusion module. This design avoids representation drift across the two modalities while allowing the fusion layer to learn how mathematical information should be incorporated into contextual representations. The fusion training data consist of (c, f) pairs, where c denotes a query or answer post and f is a formula extracted from that post.

Since a single post may contain multiple formulas, we consider two formula-selection strategies: (1) using only the longest formula in the post, or (2) randomly sampling up to six formulas from the post. Each extracted formula is paired with the corresponding post to form (c, f_i) instances. At inference time, under the multi-formula setting, each formula in a post is fused with the same contextual embedding and scored independently. The final retrieval score is obtained by averaging the similarity scores across all fused representations.

4 Experiments

4.1 ARQMath Dataset and Evaluation Setup

We evaluate MaRF on the ARQMath-3 benchmark (Mansouri et al., 2020a, 2021b, 2022a), one of the most widely used benchmarks for mathematical information retrieval. The corpus consists of mathematics-related question–answer threads from Math Stack Exchange spanning 2010 to 2018, together with metadata such as tags, upvotes, and

Model	nDCG′	MAP′	P′@10
Structure Search			
MathDowsers / L8_a018 (Kane et al., 2022)	0.474	0.164	0.247
Approach0 / a0porter (Zhong et al., 2022a,b)	0.397	0.159	0.271
MSM / BM25_system (Geletka et al., 2022)	0.396	0.122	0.194
TF-IDF(Terrier) (Mansouri et al., 2022a)	0.272	0.064	0.124
DPRL / SVM-Rank (Mansouri et al., 2022b)	0.283	0.067	0.101
Dense Retriever			
TU_DBS / base_10 (Reusch et al., 2022)	0.423	0.154	0.228
TU_DBS / RoBERTa (Reusch et al., 2022)	0.413	0.150	0.226
Approach0 / colbert (Zhong et al., 2022a)	0.418	0.162	0.251
Ensemble			
Approach0 / fusion_alpha05 (Zhong et al., 2022a)	0.508	0.216	0.345
MIRMU / MiniLM + RoBERTa (Geletka et al., 2022)	0.498	0.184	0.267
MSM / Ensemble_RRF (Geletka et al., 2022)	0.504	0.157	0.241
LLM			
GPT-4 (Satpute et al., 2024)	0.486	0.219	0.374
LLaMa-2 7B (Mansouri and Maarefdoust, 2024)	0.608	-	-
Baseline			
Fine-tuned BERT (Devlin et al., 2019)	0.436	0.182	0.262
Formula Encoder	0.392	0.151	0.253
Qwen3-Embedding 0.6B (Zhang et al., 2025)	0.583	0.261	0.405
Mathematical Representation Level Fusion (MaRF)			
Fine-tuned BERT + Formula Encoder	**0.535**	**0.221**	**0.352**
Qwen3-Embedding 0.6B + Formula Encoder	**0.622**	**0.294**	**0.433**

Table 1: ARQMath-3 Task 1 Answer Retrieval performance of different models. MaRF and formula encoder are both under Passage-level scoring and Multi-Formula input settings

author information. Mathematical expressions are originally provided in MathML and are converted to LaTeX for our preprocessing and training pipeline. Since user-written LaTeX formulas often do not strictly follow standard syntax, we apply specialized preprocessing and normalization, as detailed in Appendix A.

We evaluate MaRF on the two official ARQMath retrieval tasks: Task 1 (answer retrieval) and Task 2 (formula retrieval). Following the benchmark protocol, we report nDCG, MAP, and Precision@10, which together measure ranking quality and top-ranked retrieval precision. Appendix B presents the retrieval algorithm, additional results on NTCIR-12 (Zanibbi et al., 2016), and analyses of model size, training cost, and inference efficiency.

4.2 Main Results on ARQMath

Task 1: Answer Retrieval. Answer retrieval aims to find answers relevant to mathematical questions. Each query consists of a question title and body, including both textual descriptions and mathematical formulas. The official test set contains 100 real-world query posts, all constructed from Math Stack Exchange questions posted in 2021. The retrieval corpus includes answer posts from 2010 to 2018 in ARQMath-3, covering 0.93M questions and approximately 2.4M associated answers. Models are required to retrieve the top 1,000 relevant answers for each query.

Table 1 compares MaRF with representative prior approaches, which can be grouped into four categories: (1) structure-based retrieval methods, which match formulas using explicit structural information; (2) dense retrieval models based on Transformer or BERT-style encoders; (3) ensemble methods, which combine multiple retrieval systems by aggregating their scores; and (4) LLM-based methods, including both re-ranking approaches based on generative LLMs and retrieval approaches using embedding-oriented LLM backbones. Ensemble systems generally achieve strong performance by combining complementary signals from multiple retrieval paradigms, while recent LLM-based methods also benefit from substantially larger model capacity.

As shown in Table 1, our fine-tuned BERT baseline performs comparably to prior BERT-based dense retrievers and remains competitive among single-system methods. The standalone formula encoder performs worse, since it relies only on formula information and does not have access to richer contextual cues. In contrast, MaRF combines the strengths of contextual and formula-specific rep-

Model	nDCG′	MAP′	P′@10
Structure Search			
Tangent-S (Mansouri et al., 2022a)	0.540	0.336	0.511
MathDowsers / latex_L8_a040 (Kane et al., 2022)	0.640	0.451	0.549
MathDowsers / L8 (Kane et al., 2022)	0.633	0.445	0.549
Approach0 / a0 (Zhong et al., 2022a)	0.639	0.501	0.615
DPRL / TangentCFT2ED (Mansouri et al., 2022b)	0.694	0.480	0.611
DPRL / MathAMR (Mansouri et al., 2022b)	0.316	0.160	0.253
Dense Retriever			
Approach0 / colbert (Zhong et al., 2022a)	0.604	0.436	0.622
In.+Out (Amador and Zanibbi, 2025)	0.701	0.505	0.597
Ensemble			
DPRL / T-CFT2TED + MathAMR (Mansouri et al., 2020b, 2022b)	0.640	0.388	0.478
Baseline			
Fine-tuned BERT (Devlin et al., 2019)	0.644	0.493	0.615
Formula Encoder	0.591	0.442	0.607
Mathematical Representation Level Fusion (MaRF)			
Fine-tuned BERT + Formula Encoder	**0.732**	**0.526**	**0.627**

Table 2: ARQMath-3 Task 2 Formula Retrieval performance of different models. MaRF and formula encoder are both under Passage-level scoring and multi-formula input settings.

resentations and substantially improves retrieval effectiveness over the BERT baseline. Among non-LLM methods, MaRF achieves the strongest overall performance and even surpasses existing ensemble systems, highlighting the effectiveness of representation-level fusion for MIR. Moreover, when combined with Qwen3-Embedding-0.6B, MaRF continues to improve over the already strong LLM-based baseline, suggesting that its effectiveness when applied to larger-scale models.

Task 2: Formula Retrieval. Formula retrieval focuses on ranking formulas by relevance to a formula query. Given a query consisting only of a formula, the system must retrieve a ranked list of relevant formula instances. The official test set contains 100 formula queries sampled from Task 1 query posts. Unlike Task 1, which searches over answer posts, Task 2 retrieves from formulas appearing in both question and answer posts from 2010 to 2018, yielding a search space of roughly 4M formulas.

As shown in Table 2, our fine-tuned BERT baseline performs competitively with most prior methods, although it remains below In+Out, the strongest dense retriever. Although Task 2 is more closely aligned with the objective of the formula encoder, the standalone formula encoder still slightly underperforms the BERT baseline, likely because it does not benefit from auxiliary contextual information, which is consistent with observations in prior work (Amador and Zanibbi, 2025). By fusing contextual and formula-specific representations, MaRF captures complementary views of mathematical semantics and substantially improves formula retrieval performance over both individual encoders and prior systems.

4.3 Ablation Study

Math-Dominated Answer Retrieval. As discussed in Section 1, a key limitation of conventional dense retrievers in MIR is their heavy reliance on textual descriptions. Because textual content often dominates mathematical documents, formula information can be underrepresented during attention-based representation learning. This limitation becomes particularly severe in math-dominated settings, where documents contain many formulas but relatively little explanatory text. To examine this effect, we compare the BERT baseline and MaRF on subsets of Task 1 queries with varying levels of mathematical dominance.

We quantify mathematical dominance using the proportion of mathematical-formula tokens among all tokens in a post. Based on this ratio, we filter query posts and evaluate retrieval performance at different formula-token proportions.

Table 3 reports results on subsets of the test set grouped by the proportion of mathematical tokens in the query. Compared with its performance on the full test set, the BERT baseline degrades progressively as the proportion of mathematical tokens increases, indicating that it struggles when textual descriptions become sparse. In contrast, MaRF remains much more stable and maintains strong

performance even when mathematical tokens account for more than 70% of the input.

Model	Formula Token Percentage (Query)	nDCG′	MAP′	P′@10
BERT	>0% (All)	0.4161	0.1619	0.2426
	>50%	0.4143	0.1616	0.2585
	>60%	0.3816	0.1526	0.2418
	>70%	0.3316	0.1461	0.2562
MaRF	>0% (All)	0.4935	0.1912	0.2945
	>50%	0.5003	0.2019	0.3013
	>60%	0.4921	0.1896	0.2922
	>70%	0.4894	0.1942	0.2938

Table 3: Ablation study of answer retrieval performance when queries contain increasing proportions of mathematical formula tokens. According to the statistics of ARQMath-3, mathematical formulas tokens account for ∼32% of the tokens per query on average.

We further analyze recall over answer posts with high proportions of mathematical tokens. As shown in Table 4, the BERT baseline exhibits the same pattern, with recall decreasing noticeably as answers become more math-dominated, whereas MaRF maintains consistently stronger performance.

Model	Formula Token Percentage (Answer)	Recall
BERT	>0% (All)	0.6032
	>50%	0.5772
	>60%	0.5716
	>70%	0.5224
MaRF	>0% (All)	0.6807
	>50%	0.6754
	>60%	0.6739
	>70%	0.6902

Table 4: Ablation study of recall when retrieving answers with increasing proportions of mathematical formula tokens. According to the statistics of ARQMath-3, mathematical formulas tokens account for ∼35% of the tokens per document on average.

We also present two representative examples to illustrate the advantage of MaRF in math-dominated retrieval. When textual descriptions are limited, the BERT baseline often retrieves answers that are topically related at a surface level but mathematically mismatched. In Figure 2, the query concerns a proof of the C_r inequality. Although the BERT baseline captures the general topic of inequality proofs, it fails to distinguish the specific mathematical relationship between the C_r inequality and the triangle inequality, and therefore retrieves an incorrect answer about the latter. Similarly, in the integral example shown in Figure 3, the BERT baseline retrieves an answer involving a structurally similar integral, but fails to distinguish $\int \frac{1}{(x^2+1)^n} dx$ from $\int \frac{1}{u^2+1} du$, which are not mathematically equivalent. By contrast, MaRF places greater emphasis on formula semantics and retrieves answers that are both topically relevant and mathematically consistent with the query. These examples highlight the importance of explicitly modeling mathematical representations in math-intensive retrieval settings.

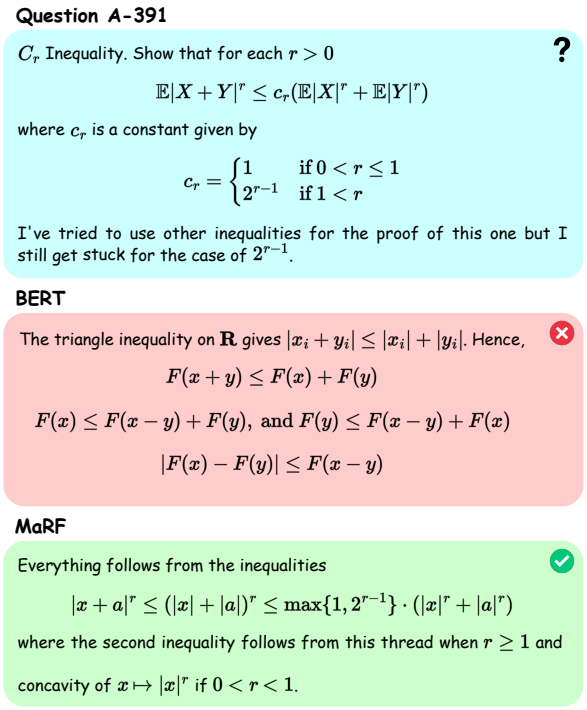

Figure 2: Retrieval comparison on question A-391. BERT captures the general inequality proof topic of the question from limited textual description, but mistakenly treat C_r inequality in question as triangle inequality problem, while MaRF effectively retrieve the correct answer.

Representation	Multi-Formula	nDCG′	MAP′	P′@10
Passage-Level	✗	0.473	0.172	0.284
Passage-Level	✓	0.498	0.195	0.297
Token-Level	✗	0.492	0.191	0.304
Token-Level	✓	**0.526**	**0.218**	**0.347**

Table 5: Ablation study on scoring strategy and multi-formula.

Effect of Scoring Strategy and Multi-Formula Fusion. Table 5 reports ablation results for two aspects of MaRF: the scoring strategy used to train the formula encoder, and the strategy used to select formulas for fusion. Token-level scor-

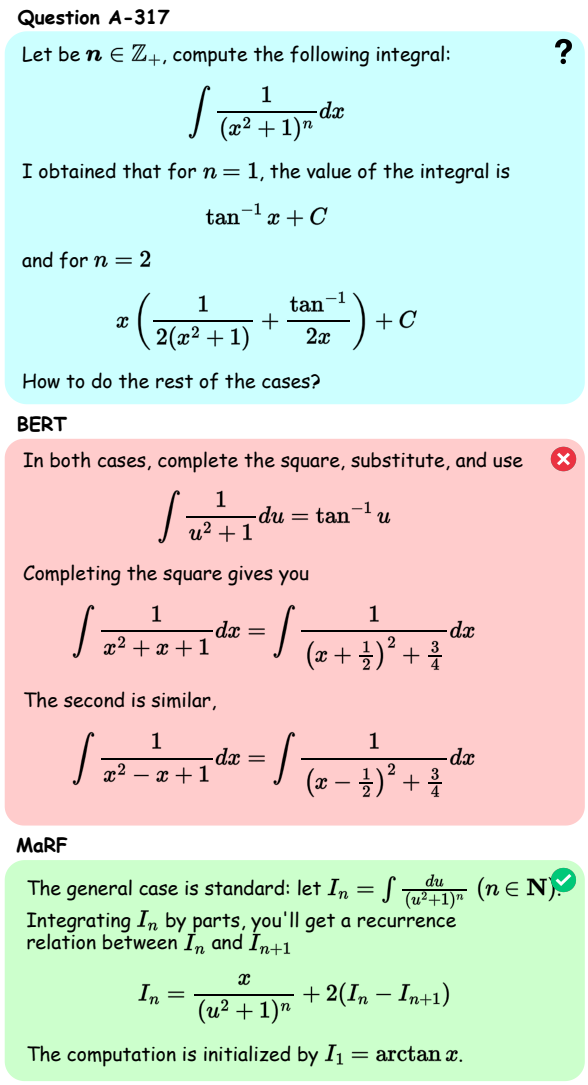

Figure 3: Retrieval comparison on question A-317. Although BERT identifies the overall integral-solving intent from sparse textual cue, but it confuses $\int \frac{1}{(x^2+1)^n} dx$ with $\int \frac{1}{u^2+1} du$. In contrast, MaRF effectively retrieve the correct answer.

ing consistently outperforms passage-level aggregation across all metrics, indicating that preserving fine-grained interactions between formula tokens is more effective than compressing an entire formula into a single vector. Passage-level aggregation tends to smooth over discriminative cues, whereas token-level matching better highlights the most relevant mathematical components. We also find that using multiple formulas from an answer post yields better results than using only the longest formula, suggesting that multiple formulas provide a more complete representation of the mathematical content in a post.

5 Conclusion

In this work, we proposed MaRF, a dual-encoder representation-level fusion framework for mathematical information retrieval. MaRF explicitly incorporates formula semantics into context-aware dense retrieval through a cross-attention-based fusion module, enabling the model to combine contextual understanding with specialized mathematical representations. Experimental results show that MaRF consistently improves retrieval performance over strong baselines and remains effective across different backbone settings. In addition, our ablation studies demonstrate that MaRF is particularly beneficial in math-dominated retrieval scenarios, where conventional dense retrievers tend to rely too heavily on textual context and under-represent mathematical content.

Limitations

Although MaRF improves MIR performance by explicitly incorporating formula-aware information into contextual retrieval representations, it still has limitations. In the current framework, mathematical semantics are modeled primarily through cleaned and normalized LaTeX representations. While effective, LaTeX is a linearized form that may not fully capture complementary structural aspects of mathematical expressions. Future work could extend MaRF to richer mathematical representations, such as Symbol Layout Trees (SLTs) and Operator Trees (OPTs), to support more comprehensive modeling of mathematical semantics.

Acknowledgments

We would like to thank the University of Illinois for its support in facilitating this research. We would also like to extend our gratitude to the National Center for Supercomputing Applications (NCSA) for providing access to high-performance computing resources.

References

Bryan Amador and Richard Zanibbi. 2025. Math formula graph retrieval using contrastive learning over visual and semantic embeddings. In *Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR)*, pages 230–237.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. In

Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), pages 3615–3620.

Pankaj Dadure, Partha Pakray, and Sivaji Bandyopadhyay. 2021. Bert-based embedding model for formula retrieval. In *CLEF (working notes)*, pages 36–46.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Dallas Fraser, Andrew Kane, and Frank Wm Tompa. 2018. Choosing math features for bm25 ranking with tangent-l. In *Proceedings of the ACM Symposium on Document Engineering 2018*, pages 1–10.

Neeraj Gangwar and Nickvash Kani. 2022. Semantic representations of mathematical expressions in a continuous vector space. *arXiv preprint arXiv:2211.08142*.

Liangcai Gao, Zhuoren Jiang, Yue Yin, Ke Yuan, Zuoyu Yan, and Zhi Tang. 2017. Preliminary exploration of formula embedding for mathematical information retrieval: can mathematical formulae be embedded like a natural language? *arXiv preprint arXiv:1707.05154*.

Martin Geletka, Vojtech Kalivoda, Michal Stefánik, Marek Toma, and Petr Sojka. 2022. Diverse semantics representation is king. In *CLEF (Working Notes)*, pages 28–39.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Emma Hamel, Hongbo Zheng, and Nickvash Kani. 2022. An evaluation of nlp methods to extract mathematical token descriptors. In *Intelligent Computer Mathematics*, pages 329–343, Cham. Springer International Publishing.

Andrew Kane, Yin Ki Ng, and Frank Wm Tompa. 2022. Dowsing for answers to math questions: Doing better with less. In *CLEF (Working Notes)*, pages 40–62.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *EMNLP (1)*, pages 6769–6781.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 39–48, New York, NY, USA. Association for Computing Machinery.

Behrooz Mansouri, Anurag Agarwal, Douglas Oard, and Richard Zanibbi. 2020a. Finding old answers to new math questions: the arqmath lab at clef 2020. In *European Conference on Information Retrieval*, pages 564–571. Springer.

Behrooz Mansouri and Reihaneh Maarefdoust. 2024. Using large language models for math information retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2693–2697.

Behrooz Mansouri, Vít Novotnỳ, Anurag Agarwal, Douglas W Oard, and Richard Zanibbi. 2022a. Overview of arqmath-3 (2022): third clef lab on answer retrieval for questions on math. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 286–310. Springer.

Behrooz Mansouri, Douglas W Oard, and Richard Zanibbi. 2020b. Dprl systems in the clef 2020 arqmath lab. In *Working notes of CLEF 2020-conference and labs of the evaluation forum*.

Behrooz Mansouri, Douglas W Oard, and Richard Zanibbi. 2021a. Dprl systems in the clef 2021 arqmath lab: Sentence-bert for answer retrieval, learning-to-rank for formula retrieval. *Proc. CLEF 2021 (CEUR Working Notes)*.

Behrooz Mansouri, Douglas W Oard, and Richard Zanibbi. 2022b. Dprl systems in the clef 2022 arqmath lab: Introducing mathamr for math-aware search. *Proc. CLEF 2022 (CEUR Working Notes)*.

Behrooz Mansouri, Shaurya Rohatgi, Douglas W Oard, Jian Wu, C Lee Giles, and Richard Zanibbi. 2019. Tangent-cft: An embedding model for mathematical formulas. In *Proceedings of the 2019 ACM SIGIR international conference on theory of information retrieval*, pages 11–18.

Behrooz Mansouri, Richard Zanibbi, Douglas W Oard, and Anurag Agarwal. 2021b. Overview of arqmath-2 (2021): Second clef lab on answer retrieval for questions on math (working notes version). *Proc. CLEF 2021 (CEUR Working Notes)*.

Bruce R Miller and Abdou Youssef. 2003. Technical aspects of the digital library of mathematical functions. *Annals of mathematics and artificial intelligence*, 38(1):121–136.

Yin Ki Ng. 2021. *Dowsing for math answers: Exploring MathCQA with a math-aware search engine*. Ph.D. thesis, University of Waterloo.

Rodrigo Nogueira, Zhiying Jiang, and Jimmy Lin. 2021. Investigating the limitations of transformers with simple arithmetic tasks. *arXiv preprint arXiv:2102.13019*.

Vít Novotnỳ, Michal Štefánik, Dávid Lupták, Martin Geletka, Petr Zelina, and Petr Sojka. 2021. Ensembling ten math information retrieval systems. CLEF.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

P Pavan Kumar, Arun Agarwal, and Chakravarthy Bhagvati. 2012. A structure based approach for mathematical expression retrieval. In *International workshop on multi-disciplinary trends in artificial intelligence*, pages 23–34. Springer.

Shuai Peng, Ke Yuan, Liangcai Gao, and Zhi Tang. 2021. Mathbert: A pre-trained model for mathematical formula understanding. *arXiv preprint arXiv:2105.00377*.

Anja Reusch, Maik Thiele, and Wolfgang Lehner. 2021a. An albert-based similarity measure for mathematical answer retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1593–1597.

Anja Reusch, Maik Thiele, and Wolfgang Lehner. 2021b. Tu_dbs in the arq math lab 2021. In *Conference and Labs of the Evaluation Forum*, pages 107–124.

Anja Reusch, Maik Thiele, and Wolfgang Lehner. 2022. Transformer-encoder and decoder models for questions on math. In *CLEF (Working Notes)*, pages 119–137.

Stephen E Robertson and Steve Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR'94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, organised by Dublin City University*, pages 232–241. Springer.

Shaurya Rohatgi, Jian Wu, and C Lee Giles. 2021. Ranked list fusion and re-ranking with pre-trained transformers for arqmath lab.

Ankit Satpute, Noah Gießing, André Greiner-Petter, Moritz Schubotz, Olaf Teschke, Akiko Aizawa, and Bela Gipp. 2024. Can llms master math? investigating large language models on math stack exchange. In *Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval*, pages 2316–2320.

Thomas Schellenberg, Bo Yuan, and Richard Zanibbi. 2012. Layout-based substitution tree indexing and retrieval for mathematical expressions. In *Document Recognition and Retrieval XIX*, volume 8297, pages 126–133. SPIE.

Yujin Song and Xiaoyu Chen. 2021. Searching for mathematical formulas based on graph representation learning. In *International Conference on Intelligent Computer Mathematics*, pages 137–152. Springer.

Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Richard Zanibbi, Akiko Aizawa, Michael Kohlhase, Iadh Ounis, Goran Topic, and Kenny Davila. 2016. Ntcir-12 mathir task overview. In *NTCIR*.

Richard Zanibbi and Dorothea Blostein. 2012. Recognition and retrieval of mathematical expressions. *International Journal on Document Analysis and Recognition (IJDAR)*, 15(4):331–357.

Richard Zanibbi, Behrooz Mansouri, Anurag Agarwal, and 1 others. 2025. Mathematical information retrieval: Search and question answering. *Foundations and Trends® in Information Retrieval*, 19(1-2):1–190.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, and 1 others. 2025. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*.

Hongbo Zheng, Suyuan Wang, Neeraj Gangwar, and Nickvash Kani. 2025. E-gen: Leveraging E-graphs to improve continuous representations of symbolic expressions. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 11772–11788, Albuquerque, New Mexico. Association for Computational Linguistics.

Wei Zhong and Jimmy Lin. 2021. Pya0: A python toolkit for accessible math-aware search. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2541–2545.

Wei Zhong, Shaurya Rohatgi, Jian Wu, C Lee Giles, and Richard Zanibbi. 2020. Accelerating substructure similarity search for formula retrieval. In *European conference on information retrieval*, pages 714–727. Springer.

Wei Zhong, Yuqing Xie, and Jimmy Lin. 2022a. Applying structural and dense semantic matching for the arqmath lab 2022, clef. In *CLEF (Working Notes)*, pages 147–170.

Wei Zhong, Jheng-Hong Yang, Yuqing Xie, and Jimmy Lin. 2022b. Evaluating token-level and passage-level dense retrieval models for math information retrieval. *arXiv preprint arXiv:2203.11163*.

Wei Zhong and Richard Zanibbi. 2019. Structural similarity search for formulas using leaf-root paths in operator subtrees. In *European Conference on Information Retrieval*, pages 116–129. Springer.

A ARQMath-3 Preprocessing

A.1 Formula Normalization

User-written LaTeX formulas often do not strictly follow canonical syntactic conventions, which makes formula representation learning challenging. To better capture the semantics and structural regularities of mathematical expressions, we apply specialized preprocessing to normalize formulas before training. Specifically, we perform the following normalization steps:

1) **Token normalization.** LaTeX provides multiple notations for expressing similar mathematical meanings. For example, `\frac` and `\dfrac` both denote fractions. We normalize such semantically equivalent symbols into a unified form.
2) **Parenthesis completion.** LaTeX formulas are often written in a non-canonical form with omitted braces or parentheses. For example, `\frac{1}{2}` may be written as `\frac12`. Since parentheses and braces are important for identifying argument boundaries, we complete omitted structures when necessary.
3) **Numerical representation.** Following prior work (Nogueira et al., 2021), we adopt a position-aware 10E-based representation for numbers. For example, 30 is represented as 3 10e 1, rather than a purely character-level form such as 3 0. This representation has been shown to improve attention over numerical expressions.

B Additional Experimental Details

B.1 Retrieval and Multi-Formula Inference

During inference, we follow the standard dense retrieval pipeline, in which similarity scores are computed independently between a query representation and candidate document representations for ranking. As discussed in Section 4.3, when a document contains multiple mathematical formulas, leveraging multiple formula representations can further improve retrieval performance.

Algorithm 1 shows the multi-formula inference procedure. Each formula extracted from a candidate document is fused with the contextual representation of that document and scored independently against the query. The final document score is obtained by averaging the scores across all fused formula-context representations.

Algorithm 1 Document Retrieval with Multi-Formula

input: query context $\boldsymbol{q}_c$, query formula $\boldsymbol{q}_f$, document context $\boldsymbol{D}_c = \{\boldsymbol{d}_{c_i}\}_{i=1}^{n}$ (n candidate documents), document formulas $\mathcal{D}_f = \{\boldsymbol{D}_{f_i}\}_{i=1}^{n}$, $\boldsymbol{D}_{f_i} = \{\boldsymbol{d}_{f_j}\}_{i=1}^{m}$ (n candidate documents, each contains m formulas)
output: retrieved top-1 document index d^*
MaRF: MaRF encoder
MaxSim: MaxSim function
$\boldsymbol{E}_q = \mathrm{MaRF}(\boldsymbol{q}_c, \boldsymbol{q}_f)$
$\boldsymbol{s} \leftarrow$ []
for all $\boldsymbol{D}_{f_i} \in \{\boldsymbol{D}_{f_1}, \ldots, \boldsymbol{D}_{f_n}\}$ **do**
  $\boldsymbol{s}_d \leftarrow$ []
  for all $\boldsymbol{d}_{f_j} \in \{\boldsymbol{d}_{f_1}, \ldots, \boldsymbol{d}_{f_m}\}$ **do**
    $\boldsymbol{E}_{d_j} = \mathrm{MaRF}(\boldsymbol{d}_c, \boldsymbol{d}_{f_j})$
    $s_j = \mathrm{MaxSim}(\boldsymbol{E}_q, \boldsymbol{E}_{d_j})$
    $\boldsymbol{s}_d \leftarrow \boldsymbol{s}_d \cup s_j$
  end for
  $s_d = \frac{1}{|\boldsymbol{s}_d|} \sum_{i=1}^{|\boldsymbol{s}_d|} s_i$
  $\boldsymbol{s} \leftarrow \boldsymbol{s} \cup s_d$
end for
$d^* = \arg\max_{|\boldsymbol{s}|}$
Return d^*

B.2 Experiment on NTCIR-12

To further evaluate the generalizability of MaRF beyond ARQMath, we conduct an additional experiment on the NTCIR-12 benchmark, a formula-only retrieval task in which both queries and candidate items are isolated LaTeX formulas. The benchmark is primarily constructed from mathematical content extracted from Wikipedia.

Model	Bpref p.	Bpref f.
GraphEmb (Song and Chen, 2021)	0.540	0.630
SciBERT (Beltagy et al., 2019)	0.363	0.512
Fine-tuned BERT (baseline)	0.546	0.613
Fine-tuned BERT + Formula Encoder	**0.584**	**0.651**

Table 6: NTCIR-12 Formula Retrieval performance of different models

As shown in Table 6, MaRF substantially improves over the BERT baseline, demonstrating that the proposed fusion framework remains effective on a different mathematical corpus and retrieval setting.

B.3 Model Size and Training Cost

The context encoder is based on a standard BERT-base backbone with approximately 110M parameters. The formula encoder is implemented as a lightweight 8-layer Transformer with 8 attention heads and a hidden size of 512, resulting in approximately 50M parameters. The fusion module consists of a cross-attention layer, a feed-forward network, and gating layers, contributing fewer than 5M parameters. In total, MaRF contains approximately 165M parameters, which is only moderately

Model	Params	GPU	Epochs	Time
BERT Encoder	∼110M	8 × A100	3	∼5h
Formula Encoder	∼50M	L40S	3	∼32h
Fusion Layer	∼5M	L40S	4	∼24h

Table 7: Training details

larger than the BERT baseline. The training time of each component is reported in Table 7.

B.4 Inference Efficiency

MaRF adopts a dual-encoder architecture, which allows document representations to be precomputed offline as a one-time indexing step. During online retrieval, only query encoding and similarity computation are required. On a single RTX 4080 GPU (16GB), MaRF processes 100 queries over a retrieval pool of approximately 1.5 million documents in around 13 minutes, with peak GPU memory usage of 11.2GB. These results suggest that the retrieval pipeline remains computationally practical and can be deployed on a single GPU.

C Additional Qualitative Examples

C.1 Additional Math-Dominated Retrieval Examples

As discussed in Section 4.3, existing dense retrievers often perform suboptimally when contextual text is limited and the proportion of mathematical content is high, largely because they rely heavily on textual descriptions. This limitation can be alleviated by incorporating complementary information from the formula encoder. In this section, we provide additional qualitative examples to further illustrate the improvements introduced by MaRF over a standalone BERT encoder.

Figure 4 shows representative ARQMath query posts that contain relatively rich textual context. In the left example, references to Euclid's lemma and Bézout's lemma, together with accompanying textual explanations, provide useful contextual cues for retrieval, allowing both models to retrieve relevant answers. Even in such cases, however, MaRF still shows an advantage. In the right example of Figure 4, both models retrieve the correct answer shown in the middle of the figure among their top-3 results, but MaRF also retrieves a second correct answer at the bottom of the figure that the BERT encoder fails to identify. This additional answer consists primarily of mathematical derivations with limited textual explanation, suggesting that MaRF better captures the underlying mathematical semantics.

When the query is dominated by mathematical formulas, the difference becomes more pronounced, as shown in Figure 5. In the left example, the BERT encoder misinterprets the positions of the base and exponent in the equation. In the right example, although the retrieved answer correctly identifies the norm operator and the variables f and g, it confuses $\|fg\|$ with $\|f - g\|$. In contrast, MaRF, by incorporating information from the formula encoder, captures the mathematical semantics more accurately and exhibits stronger robustness in math-dominated retrieval settings.

Question A-339

Extension of Euclid's lemma
This is a somewhat obvious fact that is intuitively obvious to me, but I haven't been able to construct a proof of it. Euclid's lemma state for p a prime and ab a product of integers (let's take everything to be positive for simplicity), if $p \mid ab$, then $p \mid a$ or $p \mid b$. This is clear, and I know how to prove it. Let's extend it somewhat. Suppose that a and b are two relatively prime integers, and we have $a \mid bc$ for some other integer c. Then $a \nmid b$, so it must divide c. This fact is obvious to me, but I can't figure out how to prove it.

Does anyone have any hints or advice? Do I need the assumption of positivity? (For my purposes at the moment, I only need them to be positive, but there is value in having the most general result possible).

EDIT: Updated attempt:
We have that a, b are relatively prime, so there exist r, $s \in \mathbb{Z}$ such that $ar + bs = 1$ by Bézout's lemma. Multiply through by c to get $arc + bsc = c$. Then $a \mid arc$ and $a \mid bsc$, so $a \mid c$.
How is that?

Question A-309

Number of solutions of equation over a finite field
I have a question regarding the number of solutions of a equation over a finite field $\mathbb{F}_p$. First of all, consider the equation $x^3 = a$ over $\mathbb{F}_p$, where p is a prime such that $p \equiv 2 \pmod 3$. The book that I'm currently reading says that this equation has exactly one solution in $\mathbb{F}_p$ for every $a \in \mathbb{F}_p$, because $\gcd(3, p-1) = 1$, but the book does not prove this. Unfortunately, this doesn't convince me enough. Is there a convincing elementary straightforward proof justifying why is this true?

BERT

Theorem. For $a, b, c \in \mathbb{Z}$, if $a \mid bc$ and $\gcd(a, b) = 1$, then $a \mid c$.
Proof. From Bézout's identity we know that there exist $u, v \in \mathbb{Z}$ such that $au + bv = 1$.
This gives us

$$c = (au + bv)c = a \cdot uc + bc \cdot v$$

The number a divides both summands, hence $a \mid c$.

The above result is called Euclid's lemma.

Corollary. For $a, b, c \in \mathbb{Z}$, if $a \mid bc$ and $\gcd(a, b) = 1$, then $a \mid c$.
We have $c = ka$ for some $k \in \mathbb{Z}$. Since $b \mid ka$ and $\gcd(a, b) = 1$, we get from Euclid's lemma that $b \mid k$. This implies $ab \mid ka = c$.

BERT & MaRF

The multiplicative group of $\mathbb{F}_p$ is cyclic of order $p - 1$.
As $p \equiv 2 \pmod 3$, this is not a multiple of three. Therefore cubing is a bijection from $\mathbb{F}_p$ to itself. Hence for each Y (p choices) there is a unique solution $X \in \mathbb{F}_p$, namely the cube root of $Y^2 - A$. Including the point at infinity gives a total of $p + 1$ solutions.

MaRF

Suppose that $\gcd(a, b) = 1$. Then we want to show that:
If $a \nmid c$, then $a \nmid bc$.
To do this, it suffices to prove the contrapositive:
If $a \mid c$, then $a \mid bc$.
Indeed, suppose that $a \mid bc$ so that there is some $k \in \mathbb{Z}$ such that $ak = bc$. Then since $\gcd(a, b) = 1$, it follows by Bézout's Lemma that there exist $s, t \in \mathbb{Z}$ such that:

$$\begin{aligned} as + bt &= 1 \\ acs + bct &= c \\ acs + (ak)t &= c \\ a\underbrace{(cs + kt)}_{\in \mathbb{Z}} &= c \end{aligned}$$

Thus, it follows that $a \mid c$, as desired. ■

MaRF

Hint Show $x \to x^3$ is a bijection via little Fermat and

$$\overbrace{3(2K+1) = 1 + 2(3K+1)}^{3J \equiv 1 \pmod{p-1}}$$

In detail: $x^{3J} = (x^{2K+1})^3 = x(x^{3K+1})^2 \equiv x \pmod p$ for $x \not\equiv 0$, prime $p = 3K + 2$.
Thus $x \to x^3$ is onto on the finite set $\mathbb{Z}/p$, so it is also 1-1, i.e., $x^3 \equiv y^3 \Rightarrow x \equiv y$.
Note: this answers the original version of your question (existence *and* uniqueness of cube roots).
Remark the exponent $J \equiv 2K + 1$ with $x^{3J} \equiv x^1 \pmod{p = 3K + 2}$ was computed via $\bmod p - 1 = 3K + 1$

$$3J \equiv 1 \iff J \equiv \frac{1}{3} \equiv \frac{-3K}{3} \equiv -K \equiv 2K + 1$$

using modular order reduction and $\bmod p : x^{p-1} \equiv 1, x \not\equiv 0$, by little Fermat.

Figure 4: Retrieval comparison on examples with relatively sufficient textual descriptions. Although both model retrieve valid answers, but in the right example, MaRF retrieves another correct answer which primarily consists of mathematical derivations in Top-3 results.

Question A-331

Finding roots of $4^x + 6^x = 9^x$ by hand
The function $f(x) = 4^x + 6^x - 9^x$ is such that $f(0) = 1 > 0, f(1) = 1 > 0, f(2) = -29$ and next $g(x) = (4/9)^x + (6/9)^x - 1 \Rightarrow f'(x) < 0$ for all real values of x. So $g(x)$ being monotonic the equation

$$4^x + 6^x = 9^x$$

has exactly one real solution. The question is whether this real root can be found analytically by hand.

Question A-371

Can I say $|f(x)g(x)| = \|fg\|$
Let $f, g : [0, 1] \to \mathbb{R}$ be continuous functions. Show that

$$\|fg\| \leq \|f\|\|g\|$$

what I have got so far:
$|f(x)| \leq \max |f(x)|$ = norm of f, $\|f\|$. $\forall x \in [0, 1]$. (Note: I have replaced supremum with maximum.)
$|f(x)||g(x)| = |f(x)g(x)| \leq \max |f(x)|g(x) = \|f\||g| \leq \max |f(x)| \max |g(x)| = \|f\|\|g\|$
$|f(x)g(x)| \leq \|f\|\|g\|$
As I have to show that $\|fg\| \leq \|f\|\|g\|$:
Can I say $|f(x)g(x)| = \|fg\|$? I'm not sure about that Because
$|f(x)g(x)| \leq \max |f(x)g(x)| = \max |f(x)| \max |g(x)|$
I feel I am missing concept to prove $|f(x)g(x)| = \|fg\|$, through which I think I finally can prove $\|fg\| \leq \|f\|\|g\|$ please if you guys could clarify.

BERT

One way to attack this problem is estimating number of real roots.
For that we can use Descarte's Rule of Signs. So, for

$$f(x) = x^9 + 3x^6 + 3x^3 - 16x + 9$$

We have 2 change of signs, hence the number of positive real roots are 0 or 2, but since we know $f(1) = 0$, therefore we have 2 positive real roots.
Now,

$$f(-x) = -x^9 + 3x^6 - 3x^3 + 16x + 9$$

We have 3 changes of signs, hence the number of negative real roots are 1 or 3
So, $f(x)$ has either 3 or 5 real roots
That's as far you can get without getting your hands dirty with ardous calculations involving estimation of coefficients as done by both Leucippus and Robert Z. But if you take your first case as $f(x)$ having 3 real roots, then you essentially land at Leucippus's answer, and you're in luck as that works, in this case.

BERT

This is eaiser than it sounds.
Note that $|\sin x - \sin y| = |\int_x^y \cos t dt| \leq |x - y|$ for all $x \leq y \in \mathbb{R}$. In particular for any $f, g \in C[0, 1]$, we have $|\sin f(t) - \sin g(t)| \leq |f(t) - g(t)|$ for all $t \in [0, 1]$, so
$\|F(f) - F(g)\| = \max_{0 \leq t \leq 1} |\sin f(t) - \sin g(t)| \leq \max_{0 \leq t \leq 1} |f(t) - g(t)| = \|f - g\|$
which implies that F is Lipschitz continuous (clearly stronger than continuous).
Thanks to Joonas who pointed out that my original answer only showed continuity at $f = 0$.

MaRF

Divide through by 4^x to get $1 + (\frac{6}{4})^x = (\frac{9}{4})^x$,
ie. $(\frac{6}{4})^2 x - (\frac{6}{4})^x - 1 = 0$.
Using the quadratic formula, we get $(\frac{6}{4})^x = \frac{1+\sqrt{5}}{2}$.
Since $\frac{6}{4}^x > 0$, we must have $(\frac{6}{4})^x = \frac{1+\sqrt{5}}{2}$, i.e. $x = \log_{3/2}(\frac{1+\sqrt{5}}{2})$.
Since this is the only solution, you just need to show that this number is irrational.
Suppose x is rational, i.e. $x = \frac{a}{b}$ for some integers a, b with $b > 0$.
Then, $(\frac{6}{4})^{a/b} = \frac{1+\sqrt{5}}{2}$, i.e. $\frac{3^a}{2^{b-a}} = (1 + \sqrt{5})^b$. Clearly, $\frac{3^a}{2^{b-a}}$ is rational.
However $(1 + \sqrt{5})^b = c + d\sqrt{5}$ for some positive integers c, d(Use the binomial theorem here).
Thus, $(1 + \sqrt{5})^b$ is irrational, a contradiction. Therefore, x is irrational.

MaRF

For every $x \in X$ we have $|g(x)| \leq \sup_{z \in X} |g(z)|$ by definition of the supremum, so for every $x \in X$ we may observe that

$$|f(x)g(x)| = |f(x)||g(x)| \leq |f(x)|(\sup_{z \in X} |g(z)|) = |f(x)|\|g\|$$

Since this is true for every $x \in X$ we may take the supremum on both sides of the equation to get

$$\|fg\| = \sup_{x \in X} |f(x)g(x)| \leq \sup_{x \in X} |f(x)|\|g\| = \|f\|\|g\|$$

Figure 5: Retrieval comparison on examples with insufficient textual descriptions. BERT misinterprets base and exponent positions in the equation in A-331 and confuses $\|fg\|$ with $\|f - g\|$ in A-371, whereas MaRF successfully retrieves relevant answers.