A Bigger Catch: Fine-Grained Curriculum Standards Alignment on the MathFish Benchmark

Xinman Liu, Mayank Sharma, Teah Shi
Stanford University
{xinman, masharma, teah2001}@stanford.edu

Abstract

Most existing math benchmarks for LLMs focus on evaluating correct solutions. In educational settings, however, it is equally important to understand whether LLMs understand the *pedagogical intent* behind math problems, beyond producing the right solution. Tagging curriculum standards is challenging for the same reason: distinguishing fine-grained standards requires understanding subtle pedagogical distinctions. In this paper, we use the MathFish benchmark, which frames alignment as multi-label prediction over 385 Common Core State Standards, to evaluate a three-stage pipeline inspired by baseline failure modes in retrieval and structural reasoning: curriculum-informed hard negatives (M1), a cross-encoder re-ranker (M2), and a ReAct agent paired with an LLM-as-a-judge critic (M3). We additionally evaluate a training-free alternative (A1) combining hybrid sparse-dense retrieval with curriculum graph reranking. M3 achieves 31.3% exact match, roughly 6.5× the three-shot GPT-4-Turbo baseline (Lucy et al., 2024). Error analysis shows that even with these improvements, the pipeline struggles with missing predictions, grade-level misalignment, and sibling confusion, reinforcing that precise curriculum alignment is a fundamentally hard problem in educational NLP.

1 Introduction

Most existing benchmarks for mathematical reasoning in language models ask one question above all else. Can the model get the right answer? Datasets such as GSM8k (Cobbe et al., 2021) and MATH (Hendrycks et al., 2021) test whether a model can solve a problem, but they rarely examine what mathematical competencies that problem actually exercises. In actual K-12 education, though, the categorization of math content matters more than it might seem. Professional curriculum reviewers spend months mapping published math problems to fine-grained pedagogical standards (Lucy et al., 2024). As LMs are increasingly deployed in classrooms for applications such as generating assessments or tutoring dialogues (Shi et al., 2026), it becomes important to understand whether they actually grasp the pedagogical structure behind math questions or just answer them correctly. The MathFish benchmark (Lucy et al., 2024) was introduced to evaluate exactly this, framing curriculum alignment as a multi-label prediction task over 385 math standards derived from the Common Core State Standards for Mathematics (Common Core State Standards Initiative, 2010). The standards follow the hierarchical structure of CCSSM and are commonly operationalized using resources from Achieve the Core (Student Achievement Partners, 2024). Each math problem is annotated with the standards it assesses, requiring models to predict the relevant labels from the full taxonomy.

This task turns out to be surprisingly **hard**. Many standards share enough surface vocabulary that telling them apart demands genuine understanding of pedagogical intent. When GPT-4 is evaluated with three-shot prompting and no hierarchy hints, exact-match accuracy reaches only 4.8% (Lucy et al., 2024). Most incorrect predictions fall on structurally nearby standards (siblings). In this paper, we present a three-stage pipeline where each stage is motivated by a specific failure mode of the previous one: off-the-shelf embeddings fail to separate pedagogically similar standards, so we train a contrastive bi-encoder with curriculum-informed hard negatives (**M1**); independent encoding misses fine-grained problem-standard interactions, so we add a cross-encoder re-ranker (**M2**); and threshold-based selection cannot reason about pedagogical intent, so we attach a ReAct agent with an LLM-as-a-judge critic (**M3**). We additionally evaluate a training-free alternative (**A1**) that replaces the learned retrieval front-end with hybrid sparse-dense retrieval and curriculum graph reranking, testing whether structured graph knowledge can substitute

Proceedings of the 21st Workshop on Innovative Use of NLP for Building Educational Applications, pages 208–220

for fine-tuning. Each stage is described in detail in Section 3.

Using M3, we achieve **31.3% exact match, roughly** 6.5× **the GPT-4 baseline** (Lucy et al., 2024), and discuss the roles of retrieval, reranking, and LLM reasoning in achieving precise alignment, as well as further challenges in performing this task. Our code is available at `https://github.com/yo-lxmmm/a-bigger-catch-mathfish`.

2 Related Work

Early research framed standards alignment as a multi-label text categorization problem. Using manually aligned benchmarks as training data, SVMs with bag-of-words features were applied to map curricular content to standards, showing feasibility but highlighting challenges from short standard texts and substantial lexical overlap between unrelated concepts (Yilmazel et al., 2007). Fine-tuning LMs such as BERT on task-specific educational text substantially improves classification accuracy (Shen et al., 2021), and sentence embedding models have been used to align educational resources with skill taxonomies in a shared embedding space (Li et al., 2024). However, off-the-shelf embeddings remain unreliable for math specifically: cosine similarities between different skill category embeddings often exceed 0.88 because problems share a homogeneous vocabulary of numbers, operations, and geometric terms (Xu et al., 2025). Together, these studies motivate task-specific representation learning as a prerequisite for reliable curriculum retrieval, which is the role M1 is designed to fill.

2.1 Retrieval and Reranking for Educational Tagging

TagRec (V et al., 2021) and TagRec++ (Viswanathan et al., 2022) are the closest prior applications of dense retrieval to education, framing question-to-taxonomy tagging as a similarity-based retrieval problem and showing that retrieval-based models outperform flat multi-class classifiers on hierarchical educational taxonomies. Standard contrastive objectives treat closely related labels as equally negative as completely unrelated ones; hierarchy-aware contrastive learning approaches such as Use All The Labels (Zhang et al., 2022) incorporate label structure directly into the loss. Our hard negative sampling strategy in M1 follows this intuition by oversampling standards nearby in the ATC graph, reflecting where real confusion occurs. However, bi-encoders encode problems and standards independently, limiting fine-grained interaction between them. Two-stage retrieve-then-rerank pipelines address this limitation: BEIR (Thakur et al., 2021) confirms that cross-encoder re-ranking achieves strong performance across diverse domains, motivating our M2 stage.

2.2 Agentic Reasoning and Graph-Augmented Retrieval

The ReAct framework (Yao et al., 2023) allows an LLM to interleave reasoning steps with tool calls, grounding decisions in external knowledge rather than parametric memory alone. Our M3 stage pairs a ReAct agent with an LLM-as-a-judge critic (Zheng et al., 2023) that prunes the candidate set before committing to a final prediction. As a training-free alternative, A1 replaces the learned retrieval front-end with a hybrid sparse-dense retriever (Luan et al., 2021) combining BM25 (Robertson and Zaragoza, 2009) with dense embeddings, reranked via GraphRAG (Edge et al., 2025). Recent work in educational retrieval (Jain et al., 2025) shows that graph-augmented retrieval generates more relationally coherent responses to curriculum-level queries than flat vector-based RAG, supporting the graph reranking design in A1.

2.3 The MathFish Benchmark

Our work builds directly on MathFish (Lucy et al., 2024), which frames curriculum alignment as multi-label prediction over 385 CCSS standards organized in a four-level hierarchy (grade, domain, cluster, standard) with 1,040 conceptual links in the Achieve the Core coherence map. Even GPT-4 with three-shot prompting achieves only 0.048 exact-match accuracy, and errors are structured rather than random: performance degrades as distractor standards become conceptually closer to the correct one, particularly within shared domains or ATC graph neighbors. This structured error pattern directly motivates our pipeline design—curriculum-informed hard negatives in M1 target the confusion clusters identified in MathFish's error analysis, and the ReAct agent in M3 is equipped with tools to explicitly traverse the ATC graph during reasoning.

3 Approach

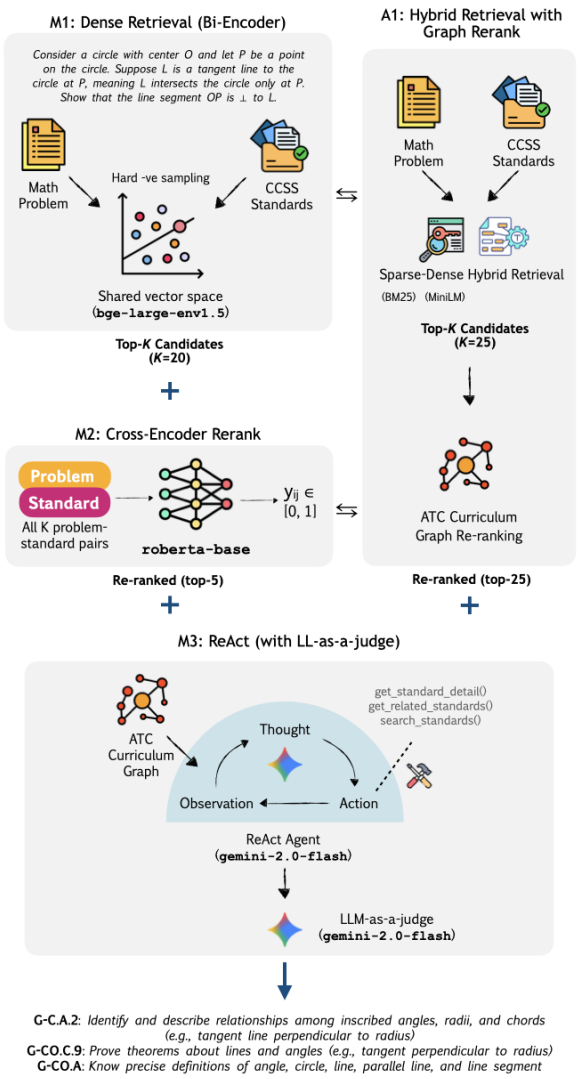

Figure 1: Overview of our pipeline. M1 (top left) trains a contrastive bi-encoder with curriculum-informed hard negatives to retrieve top-20 candidate standards from a shared embedding space. M2 (middle left) re-ranks the top-20 candidates using a `roberta-base` cross-encoder that jointly scores each problem-standard pair. A1 (top right) is a training-free alternative that replaces M1 and M2 with sparsedense hybrid retrieval followed by ATC curriculum graph reranking. All pipelines pass their top candidates to M3 (bottom), a ReAct agent powered by `gemini-2.0-flash` that iteratively reasons over standard descriptions and curriculum graph neighbors, followed by an LLM-as-a-judge critic that prunes the final prediction set.

3.1 Contrastive Bi-Encoder with Hard Negatives (M1)

For our first stage, we frame standard alignment as retrieval over 385 candidates without hierarchy hints, and adopt a bi-encoder architecture for its ability to precompute all standard embeddings and retrieve efficiently. The encoder uses `BAAI/bge-large-en-v1.5` (335M parameters) as a shared transformer backbone, with separate 256-dimensional linear projection heads for problems and standards. All embeddings are L2-normalized and scored by dot product. Given a problem embedding $\mathbf{q} = g_p(\text{enc}(x))$, a positive standard embedding $\mathbf{s}^+$, and n hard negatives $\{\mathbf{s}_j^-\}_{j=1}^n$, the model minimizes the InfoNCE contrastive loss

$$\mathscr{L}_{\text{bi}} = -\log \frac{\exp(\mathbf{q}\cdot\mathbf{s}^+/\tau)}{\exp(\mathbf{q}\cdot\mathbf{s}^+/\tau) + \sum_{j=1}^n \exp(\mathbf{q}\cdot\mathbf{s}_j^-/\tau)} \quad (1)$$

where τ is the temperature parameter controlling the sharpness of the similarity distribution. What distinguishes our approach is **curriculum-informed hard negative sampling**. MathFish's error analysis reveals that model confusions follow predictable structural patterns along the ATC hierarchy. We therefore construct hard negatives that reflect where real confusion occurs. For each (problem, positive standard) pair, we sample $n = 8$ negatives from the curriculum graph following a fixed ratio of 40% siblings (same cluster), 30% conceptual neighbors (ATC coherence links), 20% grade-adjacent standards (± 1 grade), and 10% random, excluding all gold standards for that problem. This ratio was chosen to approximate the distribution of structured confusions reported in the MathFish error analysis (Lucy et al., 2024), where sibling and conceptual-neighbor errors account for the majority of misclassifications. At inference, each problem is encoded and scored against all 385 precomputed standard embeddings, returning the top-k candidates.

3.2 + Cross-Encoder Re-Ranking (M2)

Our second stage builds on the bi-encoder (Section 3.1) by adding a cross-encoder re-ranker that jointly processes each (problem, standard) pair through full bidirectional attention. While the bi-encoder retrieves candidates efficiently, it encodes problems and standards independently, limiting the depth of interaction between them. The cross-encoder addresses this, returning the top-n re-ranked candidates as the final prediction.

We use `roberta-base` (125M parameters) with a binary classification head that outputs a relevance score $\hat{y}_{ij} \in [0,1]$ for each candidate pair (x_i, s_j). For each of the bi-encoder's top-k candidates ($k = 20$),

the problem text and standard description are concatenated into a single input sequence and passed through the full transformer. The model is trained with binary cross-entropy loss

$$\mathscr{L}_{\text{re}} = -\frac{1}{|P|} \sum_{(x_i, s_j) \in P} [y_{ij} \log \hat{y}_{ij} + (1 - y_{ij}) \log(1 - \hat{y}_{ij})] \quad (2)$$

where $y_{ij} \in \{0, 1\}$ indicates whether standard s_j is a gold alignment for problem x_i. Training pairs come from the trained bi-encoder's top-20 predictions on the training set; gold standards that appear in this candidate list serve as positives, and the rest as negatives. At inference, the cross-encoder scores all 20 bi-encoder candidates per problem, re-ranks them by score, and returns the top-n ($n = 5$) as the final prediction. The two-stage pipeline separates the retrieval task, which must search efficiently across all 385 standards, from the re-ranking task, which can afford deeper cross-attention over a focused candidate set.

3.3 + ReAct with LLM-as-a-judge (M3)

Our third stage builds directly on the first two (Sections 3.1 and 3.2), reusing the trained bi-encoder and cross-encoder as the retrieval and reranking front-end, then attaching a ReAct-based LLM agent (Yao et al., 2023) and an LLM-as-a-judge critic (Zheng et al., 2023) to the cross-encoder's top-5 output. This replaces threshold-based top-n selection with deliberative multi-step reasoning over the candidate set, addressing the limitation that the retrieve-then-rerank pipeline lacks the capacity to identify the precise gold set.

Agent. The cross-encoder's top-5 candidates are passed as the initial candidate list to a ReAct agent (Yao et al., 2023) powered by `gemini-2.0-flash-001` on Vertex AI (see Appendix A.1). The agent reasons over the candidate set through repeated THOUGHT → ACTION → OBSERVATION cycles, up to a maximum of 12 steps per problem. At each cycle, the agent assesses whether the current candidates sufficiently capture the problem's mathematical intent before invoking one of three tools: `get_standard_detail` to retrieve a standard's full description; `get_related_standards` to explore sibling and conceptual neighbors in the curriculum graph; or `search_standards` to issue a new lexical query when candidates appear misaligned. Each tool response is returned as an observation, informing the next reasoning step. Once the agent determines a final alignment, it ends with `Final Answer: ⟨standard IDs or none⟩`.

Critic. An LLM-as-a-judge critic (Zheng et al., 2023) powered by the same model (`gemini-2.0-flash-001`) prunes the predicted standard set (see Appendix A.2). Given the problem and the agent's predictions with their full descriptions, the critic returns the smallest subset that directly matches the mathematical skills being assessed, or `none` if no candidate clearly aligns. The final prediction is formed by intersecting the critic's output with the agent's original set, guarding against hallucinated identifiers.

3.4 Hybrid Retrieval with Graph Reranking and ReAct Agent (A1)

We also explore a training-free alternative (A1) which replaces the trained bi-encoder and cross-encoder with a training-free hybrid retriever and curriculum graph reranker while reusing the same ReAct agent and critic from Section 3.3, serving as a training-free comparison to the pipelines above.

Retrieval and reranking. Standards are indexed via a sparse-dense hybrid retriever (Luan et al., 2021) combining BM25 (Robertson and Zaragoza, 2009) and `all-MiniLM-L6-v2`, returning $k = 25$ candidates per problem. These are reranked by assigning each candidate a base score $\frac{1}{1+\text{rank}}$ and a coherence bonus computed against the top-5 retrieved candidates as seeds. For each (candidate, seed) pair we apply a priority cascade consistent with the ATC hierarchy, with weights reflecting the relative strength of each proximity signal: $+0.10$ for a shared CCSSM cluster (stopping further seed comparisons), $+0.05$ for a shared domain code, or $+0.03$ for a shortest path of ≤ 2 hops on the undirected ATC coherence graph, with domain and graph bonuses accumulating across seeds. All 25 reranked candidates are then passed to the ReAct agent and critic, providing a broader pool than M3's top-5. This architecture was selected after evaluating several retrieval variants on a 100-problem development subset (Appendix B), where graph reranking over hybrid-retrieved candidates outperformed sparse, hybrid, and graph-first retrieval-only alternatives.

4 Experiments

4.1 Dataset

We use MathFish (Lucy et al., 2024)[1], a dataset of approximately 21,776 math problems (13,065 train

[1] https://huggingface.co/datasets/allenai/mathfish

/ 4,356 dev / 4,355 test) drawn from Illustrative Mathematics and Fishtank Learning curricula, each labeled with fine-grained Common Core State Standards (CCSS). The 385 standards are organized hierarchically across four levels (grade → domain → cluster → standard), with 1,040 conceptual connections between them and natural language descriptions provided for each. Using ATC standard metadata[2], we filter to problems containing at least one *Addressing* or *Alignment* label, yielding 5,956 training, 1,942 development, and 2,025 test problems. Following the evaluation protocol of the original MathFish paper (Lucy et al., 2024), which reports all results on its evaluation set rather than a held-out test split, we report on the development set throughout to enable direct comparison with the baseline. The bi-encoder and cross-encoder were trained exclusively on the filtered training set; the A1 variant selection (Appendix B) used a separate 100-problem subset of the development data, and no other hyperparameters were tuned on this set. An example problem and its associated standard labels is shown in Appendix C. Label distribution is skewed: across the 584 standards with at least one training example, n_s ranges from 1 to 139 (median 11, mean 15.0). 20.7% of standards have fewer than 5 training examples, and 44.0% have fewer than 10, reflecting the uneven coverage of K-12 curricula across open educational resources (see Appendix D for the full distribution).

4.2 Evaluation

We evaluate using the following metrics: *Exact Match* requires the predicted standards to exactly match gold; *Weak Accuracy* requires at least one overlap. *Micro/Macro F1* capture token-level and standard-level agreement. *Ret* and *Rerank* report recall@5 before and after reranking (cross-encoder for M2/M3, curriculum graph for A1). *GraphDist* measures the average minimum graph distance between predicted and gold standards in the ATC graph (↓ better), and *SibConf* measures the fraction of errors landing on a same-cluster sibling (↓ better). *Avg Pred* tracks the average number of predicted standards per problem.

4.3 Experimental Details

Bi-Encoder (M1). We initialize from `BAAI/bge-large-en-v1.5` (335M parameters) and train on 5,956 problems using AdamW (learning rate 2×10^{-5}, batch size 16) for 15,000 gradient steps with temperature $\tau = 0.03$ and $n = 8$ curriculum-aware hard negatives per positive. The best checkpoint is selected by minimum validation loss. Training uses FP16 on a single NVIDIA H100 (80 GB) and completes in approximately 50 minutes.

Cross-Encoder Re-Ranker (M2). We fine-tune `roberta-base` (125M parameters) for 3 epochs using AdamW (learning rate 2×10^{-5}, batch size 32, max sequence length 256). Training pairs are constructed by running M1 on the training set with $k = 20$, yielding 119,120 scored pairs per epoch and 11,169 gradient steps total. Training completes in approximately 20 minutes on the same H100.

ReAct Agent and LLM-as-a-Judge (M3). M3 requires no training and operates at inference time. Both the ReAct agent and LLM-as-a-judge use `gemini-2.0-flash-001`.

Hybrid Retriever and Graph Reranker (A1). A1 also requires no training. BM25 sparse retrieval is combined with dense embeddings from `all-MiniLM-L6-v2` and candidates are re-ranked using the ATC curriculum graph via GraphRAG before being passed to the same agentic reasoning mechanism as M3.

4.4 Results

Table 1 presents the main results for all pipeline stages and the training-free alternative against the results of three-shot GPT-4-Turbo under self-guided tree traversal reported in the original MathFish paper (Lucy et al., 2024) as the baseline; all evaluated on the same development set (Addressing/Alignment labels only).

The curriculum-aware contrastive bi-encoder (M1) achieves strong retrieval coverage with 20 standards per problem. Adding the cross-encoder re-ranker (M2) tightens the prediction set to 5 candidates and cuts graph distance from 4.24 to 2.42, confirming that cross-attention over concatenated problem-standard pairs captures alignment signals that independent embeddings miss. Both M1 and M2 reach 0.000 exact match despite strong weak accuracy, which is expected given that predicting 20 and 5 candidates respectively against a gold average of 1.47 makes exact match nearly impossible by construction—these are fixed-size prediction sets, not threshold-based ones. A fairer evaluation of reranking alone would apply a score threshold to M2's cross-encoder outputs to produce a variable-size prediction set; we leave this to future work and note that the single-pass critic (28.4% EM)

[2] https://huggingface.co/datasets/allenai/achieve-the-core

Table 1: Main results on the full Addressing/Alignment-filtered development set (1,942 problems). – denotes metrics not reported.

	Accuracy				Recall@5		Graph Quality		
Method	**Exact**	**Weak Acc**	**Micro F1**	**Macro F1**	**Ret**	**Rerank**	**GraphDist↓**	**SibConf↓**	**Avg Pred***
Baseline: Three-Shot GPT-4-Turbo†	0.048	0.502	–	–	–	–	1.90	–	3.05
M1: Biencoder	0.000	**0.728**	0.088	0.087	**0.632**	–	4.24	**0.020**	20.00
M2: Biencoder + Cross Rerank	0.000	0.688	0.266	0.259	**0.632**	**0.688**	2.42	0.060	5.00
M3: Biencoder + Cross Rerank + ReAct	**0.313**	0.589	**0.455**	**0.478**	**0.632**	**0.688**	**0.93**	0.109	1.37
A1: Hybrid + Graph Rerank + ReAct	0.275	0.541	0.401	0.432	0.271	0.311	1.047	0.100	**1.438**

†Three-shot GPT-4-Turbo under self-guided tree traversal, as reported in the original MathFish paper (Lucy et al., 2024); evaluated on the same development set (Addressing/Alignment labels only).
*Gold average = 1.47 for the A+A filtered set.

provides a practical estimate of what M2's candidate pool can support with a downstream selector. This confirms that retrieval and reranking alone are insufficient without a mechanism to identify the precise gold set and reason over pedagogical intent.

As demonstrated in M3, attaching the ReAct agent and critic to M2's top-5 candidates produces the largest gain in the ablation: exact match rises from 0.000 to 0.313, roughly 6.5 times the three-shot GPT-4-Turbo baseline (0.048) (Lucy et al., 2024). The agent's multi-step reasoning (e.g., inspecting standard descriptions, exploring graph neighbors, and invoking the critic to prune over-predictions) converts a ranked list into a substantially more precise prediction set, reducing average predictions from 5.00 to 1.37 (close to the gold average of 1.47) and achieving the best Micro-F1 (0.455) and Macro-F1 (0.478) across all stages and the alternative. M3 also achieves the best graph distance at 0.93, suggesting that the cross-encoder's structurally tightened candidate pool effectively guides the agent toward predictions that are close in the curriculum graph even when they do not exactly match gold.

To situate the contribution of fine-tuned retrieval, we compare M3 against A1. Despite passing a larger reranked candidate pool to the agent (top-25 vs. top-5), A1 achieves lower exact match (0.275 vs. 0.313) and higher graph distance (1.047 vs. 0.93). The performance gap is further reflected in recall at the top-5 candidate level: A1's untuned hybrid retriever achieves retrieval recall of only 0.271, which improves marginally to 0.311 after graph reranking. In contrast, M3's fine-tuned bi-encoder achieves retrieval recall of 0.632, preserved and improved to 0.688 after cross-encoder reranking. We note a confound in this comparison: A1 passes 25 candidates to the agent while M3 passes only 5, meaning the gap could reflect lower retrieval recall, a larger and noisier candidate pool confusing the agent, or both. We leave a controlled ablation varying candidate pool size to future work, but note that A1's retrieval recall at top-5 (0.311) is already substantially lower than M3's (0.688), suggesting retrieval quality is the primary driver.

To further isolate the contribution of the ReAct agent's iterative tool use from simply having a zero-shot pruning LLM call, we evaluated a zero-shot single-pass critic applied directly to M2's top-5 using the same model and prompt as M3. This ablation achieves 28.4% exact match in 48 minutes (vs. 2.64 hours for M3), indicating that the trained retrieval pipeline (M1 and M2) accounts for the majority of the improvement over the prompting-only baseline, with the ReAct loop and tool use contributing a further +2.9 percentage points and meaningfully tighter graph distance (0.929 vs. 0.981) at the cost of additional inference time.

We note that M3 uses `gemini-2.0-flash-001` while the baseline uses GPT-4-Turbo, confounding pipeline design with model choice. The A1 pipeline uses the same Gemini backbone as M3 yet achieves only 0.275 exact match vs. M3's 0.313, and the single-pass critic ablation with Gemini already reaches 0.284 without any ReAct loop, together suggesting the retrieval pipeline contributes independently of the LLM backbone. The D* ablation (Appendix B) further confirms that model choice matters within our framework. We acknowledge that cleanly isolating the pipeline contribution would require holding the backbone constant, which we leave to future work.

In terms of computational cost, over the full 1,942-problem development set, M3 completes in 2.64 hours (4.89 s/problem), averaging 4.00 Gemini API calls per problem (approximately 3 ReAct steps plus 1 critic call); A1 is slightly slower at 5.38

s/problem (2.90 h total), averaging 4.54 calls per problem. Approximate API cost is $2.00-$2.50 for M3 and $3.00-$3.50 for A1 across the full development set, suggesting both pipelines are deployable at practical scale.

5 Analysis

To better understand where and why our systems succeed and fail, we analyze performance across curriculum strata, error patterns, and system behaviors beyond aggregate metrics.

Performance across curriculum strata. Examining the results for M3 (the best-performing configuration), performance varies substantially across grade levels and domains (Tables 5 and 6 in Appendix E). Exact match is strongest at early elementary grades (K-2: 0.27-0.30) and decreases toward upper elementary and secondary levels, reaching its lowest at grade 5 (0.12) and high school (0.13). Domain-level results show similar variance: Geometry (0.30) and Number & Operations in Base Ten (0.32) achieve the strongest performance, while Functions (0.09) and Fractions (0.10) the weakest. Such variance is consistent with the original MathFish paper which reports exact accuracy ranging from 0.151 (Counting & Cardinality) to 0.011 (Functions) for GPT-4 (Lucy et al., 2024), suggesting domain difficulty reflects intrinsic properties of the task. These grade- and domain-level patterns are reflected in the systematic failure modes we characterize below.

Label frequency and performance. To disentangle whether performance gaps reflect pedagogical difficulty or data sparsity, we examine the relationship between per-standard training frequency and M3 exact match (see Appendix D). At the problem level, we find a modest but significant correlation between training frequency of the rarest gold standard and exact match ($\rho = 0.234$, $p < 10^{-24}$): problems whose gold standards appear rarely in training (Q1, $n_s \leq 10$) achieve 14.8% EM, compared to 44.9% for problems with frequently-occurring standards (Q4, $n_s \geq 31$). However, when examined at the per-standard level directly, correlating each standard's training frequency with its conditional exact-match rate across development problems, the relationship disappears entirely ($\rho = -0.016$, $p = 0.894$, over 76 standards with $\geq$10 development problems). This dissociation suggests that the problem-level frequency effect reflects the intrinsic difficulty of problems that happen to target rare standards, rather than the model simply failing due to insufficient training signal for those standards. Pedagogical subtlety and data sparsity are thus partially confounded in this benchmark, and both likely contribute to the performance gaps we observe.

Error analysis. Qualitative error analysis of M3's failure cases reveals three major error patterns. First, *incomplete prediction*: the system often identifies a primary standard but misses additional relevant ones. For example, for a problem asking students to graph a quadratic function and annotate its features, the agent predicts `F-IF.B.4` (interpret key features of graphs and tables, and sketch graphs showing key features given a verbal description of the relationship) but misses `F-IF.C.7a` (graph linear and quadratic functions and show intercepts, maxima, and minima), despite the problem explicitly requiring graphing these elements. This pattern suggests the agent stops after identifying the most salient standard without verifying completeness. It also triangulates with the low exact match at high school (0.13) where problems frequently align to multiple standards (Table 5 in Appendix E). A targeted fix for future work could be an explicit verification prompt after finalization: *"Are there additional standards that also apply? Check siblings and related standards before finalizing."*

Second, *grade-level misalignment*: the system predicts conceptually related standards at the wrong grade level. For example, a problem presenting a linear relationship $y = 3x + 6$ and asking students to notice patterns falsely received the high school standard prediction `F-IF.B.4` (interpret key features of graphs and tables given a function modeling the relationship between two quantities), rather than the 8th grade standard `8.F.B.4` (construct a function to model a linear relationship from a table). Because the system lacks explicit grade-level calibration, and that standards across grades often share closely overlapping descriptions due to the spiral nature of math curricula (where later standards build upon earlier ones) (Ireland and Mouthaan, 2020), it fails to recognize that `8.F.B.4` emphasizes constructing linear functions from tables as an introductory concept, while `F-IF.B.4` involves more sophisticated interpretation of more diverse functions and relationships. Future work could incorporate grade-level filtering or an explicit complexity classification step.

Third, *sibling confusion*: the system still con-

flates structurally adjacent but pedagogically distinct standards within the same cluster. For instance, for a problem asking which scenarios require multiplying $\frac{1}{8} \times \frac{2}{5}$, the agent predicts `5.NF.B.4` (apply and extend previous understandings of multiplication to multiply fractions) when the gold includes `5.NF.B.6` (solve real world problems involving multiplication of fractions and mixed numbers). In this case, while both standards share the same parent cluster (`5.NF.B`), there is a distinction between *applying* the operation versus *solving* contextualized problems requiring it. This validates the MathFish paper finding that models predict labels that are "close to ground truth, but differ in subtle ways" (Lucy et al., 2024), and also explains the weak performance on Functions (0.09) and Fractions (0.10) as domains with dense clusters of similar-sounding standards. A targeted improvement could be to augment the `get_standard_detail` tool of the ReAct agent with contrastive descriptions for common confusion pairs.

Implications for practitioner workflows. The pipeline offers two natural operating points for educators. M2's top-5 ranked list (weak accuracy 0.688) supports a *suggest-and-verify* workflow in which a reviewer scans a short candidate set rather than searching across all 385 standards manually. M3's variable-size output (exact match 0.313, weak accuracy 0.589, graph distance 0.93) supports a *draft-tagging* workflow in which the system proposes an initial alignment that humans then audit. Because M3's graph distance of 0.93 means that even incorrect predictions typically land within the same cluster or an adjacent one, the expected correction effort is a local adjustment (e.g., swapping a sibling standard) rather than a full re-search of the taxonomy. From an instructional design perspective, not all error types carry equal cost: grade-level misalignment is arguably the most consequential because it mistargets developmental appropriateness, whereas sibling confusion preserves the correct instructional focus area, and incomplete prediction falls between the two since the primary standard is usually identified. These distinctions suggest that future deployment should surface confidence signals alongside predictions so that reviewers can prioritize verification effort accordingly.

Curriculum alignment as a fundamentally hard problem. These results underscore that curriculum alignment is harder than mathematical reasoning. Solving a problem requires producing a correct answer, but aligning it to standards requires inferring the pedagogical intent behind its construction, the specific competencies it targets, and how those relate to structurally adjacent standards in the curriculum hierarchy (Lucy et al., 2024; Sonkar et al., 2024). The spiral nature of math curricula (Ireland and Mouthaan, 2020), where later standards build upon earlier ones with overlapping descriptions, further compounds this difficulty. This explains why embedding-based retrieval and reranking is insufficient even with strong fine-tuned encoders (Xu et al., 2025) and why even our best system with agentic reasoning capabilities achieves only 31.3% exact match. Nonetheless, it is worth noting that out of the cases with identified errors, more than half of them achieve weak match, and M3's graph distance of 0.93 already substantially improves over GPT-4's 1.90 (Lucy et al., 2024), suggesting the system is frequently in the right neighborhood. This raises the question of whether exact match is the right criterion in the first place: curriculum standards alignment requires judging whether a problem enables students to learn the "full intent" of a standard—a nuanced judgment that Lucy et al. (2024) suggest professional curriculum reviewers may not always make consistently, though the degree of inter-annotator disagreement remains unquantified in the literature. Future work should move beyond binary exact match toward evaluation protocols that assign partial credit weighted by curriculum proximity, and measure human-human and human-AI agreement rather than accuracy against a single gold standard, better reflecting the inherently subjective nature of curriculum alignment.

6 Conclusion

We studied automated curriculum alignment on the MathFish benchmark, testing whether training-based retrieval and agentic reasoning can improve upon prompting-only baselines. Our ablation shows that contrastive retrieval (M1) and cross-encoder re-ranking (M2) provide strong candidate coverage but cannot predict exact matching curriculum standard sets on their own, both producing 0.000 exact match. This is expected given that their prediction sets are fixed at 20 and 5 candidates respectively, far exceeding the gold average of 1.47. Attaching a ReAct agent and LLM-as-a-judge critic (M3) raises exact match to 0.313,

approximately 6.5 times the GPT-4-Turbo baseline (Lucy et al., 2024). Notably, a single-pass LLM-as-a-judge critic without the ReAct loop already achieves 28.4%, reflecting that M1+M2's role is to transform the 385-way alignment problem into a focused pruning task over a small, high-quality candidate set, in contrast to the prompt-only baseline (Lucy et al., 2024), which requires the LLM to navigate the full taxonomy unaided. The ReAct loop's iterative tool use then adds a further +2.9 percentage points on top.

Limitations

As demonstrated, even our best system achieves only 31.3% exact match, and error analysis reveals three systematic failure modes: incomplete prediction, grade-level misalignment, and sibling standards confusion. These failures are partly structural: the spiral nature of math curricula means that standards across grades deliberately share overlapping language, making grade-level disambiguation fundamentally harder than lexical similarity would suggest. The system also has no mechanism to reason about the cognitive demand or instructional context of a problem, both of which human reviewers draw on when distinguishing, for example, a standard targeting procedural fluency from one targeting conceptual understanding within the same cluster. Additionally, the 40/30/20/10 hard negative sampling ratio in M1 is a fixed heuristic informed by prior error analysis rather than an empirically optimized hyperparameter, and we leave ablation of this ratio to future work. A related concern is that sibling standards sampled as hard negatives may occasionally be true positives for a given problem, potentially teaching the bi-encoder to over-suppress genuinely related standards and contributing mechanistically to the incomplete prediction failure mode we identify in Section 5. Quantifying and filtering contaminated negatives is an important direction for future work. Finally, the agent and critic in M3 share the same underlying model, limiting critic independence: while the asymmetric prompting—the agent explores, the critic prunes—produces empirically observable reduction in predictions (5.00 to 1.37), systematic errors rooted in the model's parametric representation of CCSS standards, such as grade-level misalignment and sibling confusion, are unlikely to be caught by a critic sharing those same representations. Future work should explore using a different model family or a trained discriminative verifier as the critic.

Beyond the model itself, exact match may not be the right evaluation criterion for this task. Curriculum alignment is inherently a judgment call: while Lucy et al. (2024) note qualitatively that professional reviewers may disagree, neither that work nor ours quantifies inter-annotator agreement directly, and establishing a human agreement ceiling remains important future work. Future work should also develop evaluation protocols that assign partial credit weighted by curriculum proximity rather than treating a single gold annotation as ground truth. On the modeling side, incorporating grade-level signals, contrastive standard descriptions for common confusion pairs, and explicit completeness verification steps could address the identified failure modes more directly.

Ethical Considerations

Curriculum alignment systems used in educational settings have real consequences for both students and teachers. Incorrect standard predictions can result in poorly targeted instruction or assessments, and may disproportionately impact students in under-resourced schools where automated tools are more likely to replace expert review. We advise against deploying these systems without human oversight. In practice, they should be co-developed with educators and curriculum specialists to ensure the outputs are pedagogically sound and interpretable, and they should be evaluated against the judgment of domain experts rather than relying solely on proposed metrics.

References

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Common Core State Standards Initiative. 2010. Common core state standards for mathematics. https://www.corestandards.org/Math/.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. 2025. From local to global: A graph rag approach to query-focused summarization. *Preprint*, arXiv:2404.16130.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and

Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*.

Jo Ireland and Melissa Mouthaan. 2020. Perspectives on curriculum design: comparing the spiral and the network models.

Amay Jain, Liu Cui, and Si Chen. 2025. Aligning llms for the classroom with knowledge-based retrieval – a comparative rag study. *Preprint*, arXiv:2509.07846.

Zhi Li, Zachary A. Pardos, and Cheng Ren. 2024. Aligning open educational resources to new taxonomies: How ai technologies can help and in which scenarios. *Computers Education*, 216:105027.

Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2021. Sparse, Dense, and Attentional Representations for Text Retrieval. *arXiv preprint*. ArXiv:2005.00181 [cs].

Li Lucy, Tal August, Rose E. Wang, Luca Soldaini, Courtney Allison, and Kyle Lo. 2024. MathFish: Evaluating language model math reasoning via grounding in educational curricula. *Preprint*, arXiv:2408.04226.

Stephen Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

Jia Tracy Shen, Michiharu Yamashita, Ethan Prihar, Neil Heffernan, Xintao Wu, Sean McGrew, and Dongwon Lee. 2021. Classifying math knowledge components via task-adaptive pre-trained bert. In *Artificial Intelligence in Education: 22nd International Conference, AIED 2021, Utrecht, The Netherlands, June 1418, 2021, Proceedings, Part I*, page 408419, Berlin, Heidelberg. Springer-Verlag.

Yuhong Shi, Kun Yu, Yifei Dong, and Fang Chen. 2026. Large language models in education: a systematic review of empirical applications, benefits, and challenges. *Computers and Education: Artificial Intelligence*, 10:100529.

Shashank Sonkar, Kangqi Ni, Sapana Chaudhary, and Richard G. Baraniuk. 2024. Pedagogical Alignment of Large Language Models. *arXiv preprint*. ArXiv:2402.05000 [cs].

Student Achievement Partners. 2024. Achieve the core. `https://achievethecore.org/`.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models. *Preprint*, arXiv:2104.08663.

Venktesh V, Mukesh Mohania, and Vikram Goyal. 2021. Tagrec: Automated tagging of questions with hierarchical learning taxonomy. *Preprint*, arXiv:2107.10649.

Venktesh Viswanathan, Mukesh Mohania, and Vikram Goyal. 2022. Tagrec++: Hierarchical label aware attention network for question categorization. *Preprint*, arXiv:2208.05152.

Qingshu Xu, Hong Jiao, Tianyi Zhou, Ming Li, Nan Zhang, Sydney Peters, and Yanbin Fu. 2025. Automated alignment of math items to content standards in large-scale assessments using language models. *Preprint*, arXiv:2510.05129.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. ReAct: Synergizing Reasoning and Acting in Language Models. *arXiv preprint*. ArXiv:2210.03629 [cs].

Ozgur Yilmazel, Niranjan Balasubramanian, Sarah Harwell, Jennifer Bailey, Anne Diekema, and Elizabeth Liddy. 2007. Text categorization for aligning educational standards. page 73.

Shu Zhang, Ran Xu, Caiming Xiong, and Chetan Ramaiah. 2022. Use all the labels: A hierarchical multi-label contrastive learning framework. *Preprint*, arXiv:2204.13207.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. *arXiv preprint*. ArXiv:2306.05685 [cs].

A Prompts for ReAct Agent with LLM-as-a-Judge

A.1 ReAct agent prompt

◇ ReAct system prompt for M3 and A1

You are an expert at aligning math problems to Common Core State Standards.

Task. Given a math problem and (optionally) an initial list of candidate standards, decide which standards the problem aligns with. A problem may align to multiple standards if it addresses multiple concepts or skills. You may use tools to:

1. `get_standard_detail(standard_id)` – fetch full description and metadata for one standard.
2. `get_related_standards(standard_id, relation)` – fetch related standards (siblings in same cluster or conceptual links; `relation` is "siblings" or "conceptual").
3. `search_standards(query)` – when the initial candidates look wrong, search again with a short phrase (e.g., "fractions with like denominators", "linear equations grade 8") to find more candidates.

Respond in this exact format at every step:

- `Thought: <your reasoning>`
- `Action: <exactly one tool call: tool_name(arg1, arg2)>`

When you are done and ready to give the final set of standard IDs:

- `Thought: <brief reasoning summarizing why these standards are correct and complete>`
- `Final Answer: <comma-separated list of standard IDs, e.g., 7.NS.A.1c, 7.EE.A.1, or none>`

Rules.

- Only output standard IDs that appear in tool results (initial candidate list, `get_standard_detail`, `get_related_standards`, or `search_standards`). Do not invent new IDs.
- A problem aligns with a standard only if students could reasonably learn the full intent of that standard from the problem, not just see a related idea in passing.
- Include all standards that the problem truly aligns with; if a problem teaches multiple concepts, list all relevant IDs.
- If you are not confident that any standard clearly meets this bar, output `Final Answer: none`.
- Use exact CCSS IDs (e.g., `7.NS.A.1c`, `A-REI.D.11`).

A.2 Critic prompt

◇ Critic prompt (LLM-as-a-judge) for pruning predictions for M3 and A1

You are an expert at aligning math problems to Common Core State Standards.

You will be given:

1. A math problem.
2. A list of *candidate* standards (ID + description) that another agent thinks might apply.

Your job is **only** to prune this candidate list:

- You may only select from the given IDs. Do *not* invent or add new standard IDs.
- A problem aligns with a standard only if it can enable students to learn the full intent of that standard's description, not just mention a related idea in passing.
- Choose the smallest subset of IDs that directly match what the problem is assessing or teaching.
- If two standards are clearly redundant (same idea, different grade bands), keep the best match and drop the others.
- If you are not confident that any candidate clearly meets this bar, prefer `none` over guessing. If none of the candidates are good matches, you may output `none`.

Respond in this format **only**:

- `Thought: <your reasoning>`
- `Final Answer: <comma-separated list of chosen IDs, or 'none'>`

B Preliminary Pipeline Variants (100-Problem Development Subset)

All variants A–F use `gemini-2.0-flash-001` with the same ReAct agent and critic, and pass top-25 retrieved candidates to the agent; only the retrieval stage (and optionally a verifier) differs. Each was evaluated on a fixed 100-problem subset of the Addressing/Alignment-filtered development set (all problems have non-empty gold labels; $N = 100$ for all rows).

A No retrieval (no RAG).

B BM25 only.

C Hybrid retrieval (BM25 + dense).

D Hybrid retrieval + ATC curriculum graph reranking (cluster and domain proximity).

E Graph-first retrieval: hybrid seeds ($k_{\text{seed}} = 3$) expanded via the ATC curriculum graph.

F Same as E, with an additional per-standard verifier (Yes/No) before the critic.

Model ablation. Variant D* replicates pipeline D but substitutes `gemini-2.5-flash-lite` for `gemini-2.0-flash-001` to ablate the effect of model choice; it is substantially slower and predicts more standards per problem on average, but modestly improves weak accuracy.

Table 2 reports results; **the selected pipeline is Variant D**.

Table 2: Preliminary results on the fixed 100-problem development subset ($N = 100$). Bold indicates best per metric. Graph Distance (GraphDist) and Sibling Confusion Rate (SiblingConf) are graph-based alignment quality metrics where lower is better.

Variant	Exact	Weak Acc	Micro F1	Macro F1	R@5	R@20	GraphDist	SiblingConf
A	0.26	0.48	0.38	0.39	–	–	1.13	**0.10**
B	0.23	0.48	0.36	0.37	0.25	0.43	1.23	0.15
C	0.28	0.59	0.42	0.46	0.37	0.69	**0.95**	0.15
D	**0.30**	**0.64**	**0.46**	**0.50**	**0.39**	**0.70**	1.13	0.15
E	0.24	0.57	0.41	0.43	0.28	0.48	1.04	0.14
F	**0.30**	0.54	0.41	0.44	0.29	0.46	1.11	0.17
D*	0.17	**0.70**	0.35	0.43	0.40	**0.70**	1.49	**0.08**

Variants A–F use `gemini-2.0-flash-001`; D substitutes `gemini-2.5-flash-lite` on the selected pipeline (D) as an ablation on model choice.

C Example from MathFish Benchmark (Lucy et al., 2024)

◇ **Example**

Problem. Consider a circle with center O and let P be a point on the circle. Suppose L is a tangent line to the circle at P, that is, L meets the circle only at P. Show that $\overline{OP}$ is perpendicular to L.

Output Standards:

- **G-C.A.2** Identify and describe relationships among inscribed angles, radii, and chords (e.g., tangent line perpendicular to radius).
- **G-CO.C.9** Prove theorems about lines and angles (e.g., tangent perpendicular to radius).
- **G-CO.A** Know precise definitions of angle, circle, line, parallel line, and line segment.

D Label Frequency Distribution

Table 3 reports the distribution of training examples per standard in the Addressing/Alignment-filtered training set. Of the 385 standards in the full CCSS taxonomy, 584 appear at least once across the 5,956 training problems.[3] Distribution is right-skewed: the median standard has 11 training examples, but 20.7% have fewer than 5 and 44.0% have fewer than 10, while the most frequent standard appears in 139 problems.

[3]The count exceeds 385 because multi-label problems increment each appearing standard, and sub-standard labels introduced via inheritance (Appendix A.2 of Lucy et al. (2024)) expand the effective label set.

Table 3: Distribution of training examples per standard (n_s) across standards with at least one training example in the Addressing/Alignment-filtered training set (5,956 problems).

Training examples (n_s)	**# standards**	**% of standards**
< 2	32	5.5%
2–4	89	15.2%
5–9	136	23.3%
10–19	174	29.8%
20–50	114	19.5%
> 50	39	6.7%
Total	584	100%

Summary statistics: min = 1, median = 11, mean = 15.0, max = 139.

Table 4 reports M3 exact match stratified by training frequency of the rarest gold standard per problem ($m = \min_{s \in G} n_s$) and the two correlation analyses described.

Table 4: M3 exact match by training frequency quartile. Problems are binned by m, the training count of the rarest gold standard in their gold set. Quartiles are equal-frequency (approximately 485–486 problems each).

Quartile	n	**EM**	**min** m	**max** m
Q1 (lowest m)	486	0.148	0	10
Q2	485	0.332	10	18
Q3	485	0.324	19	31
Q4 (highest m)	486	0.449	31	139
Overall	1,942	0.313	–	–

Spearman ρ between $\log(1+m_i)$ and exact match (per problem, $n = 1{,}942$): $\rho = 0.234$, $p = 1.31 \times 10^{-25}$. Spearman ρ between $\log(1+n_s)$ and conditional exact-match rate (per standard, $n = 76$ standards with $\geq$10 dev problems): $\rho = -0.016$, $p = 0.894$.

E M3 Performance by Grade and Domain

Tables 5 and 6 report exact match disaggregated by grade level and domain for M3 (BiEncoder + Cross-Encoder Rerank + ReAct) on the full Addressing/Alignment-filtered development set ($N = 1942$).

Table 5: Exact match by grade level on the full Addressing/Alignment-filtered development set. N counts problem–grade pairs.

	K	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**HS**
N	166	221	211	317	326	340	396	375	387	1027
M3: BiEncoder + Cross + ReAct	0.27	0.29	0.30	0.26	0.22	0.12	0.19	0.21	0.22	0.13

Table 6: Exact match by domain on the full Addressing/Alignment-filtered development set. N counts problem-domain pairs.

	C&C	**Func**	**Geom**	**M&D**	**NOBT**	**NOF**	**NSQ**	**O&A**	**R&P**	**S&P**
N	72	407	646	230	465	254	214	914	217	347
M3: BiEncoder + Cross + ReAct	0.19	0.09	0.30	0.23	0.32	0.10	0.19	0.18	0.16	0.24

*Note: C&C = Counting & Cardinality, Func = Functions, Geom = Geometry, M&D = Measurement & Data, NOBT = Number & Operations in Base Ten, NOF = Number & Operations – Fractions, NSQ = Number Systems & Quantity, O&A = Operations & Algebra, R&P = Ratios & Proportional Relationships, S&P = Statistics & Probability.