AI-Powered Olympiad Tutor: Democratizing Access to Mathematical Olympiad Training

Niranjan Naik[1], Rajdeepsinh Jadeja[2], Ram Palan[3], Neil Rahate[4], and Dr. Abhijit Joshi[5]

[1,2,3,4,5]Dwarkadas J. Sanghvi College of Engineering, Mumbai, India

[1]niranjannaik4045@gmail.com, [2]rajdeepjadeja2806@gmail.com, [3]ramrampalan@gmail.com, [4]neilrahate4@gmail.com, [5]abhijit.joshi@djsce.ac.in

***Abstract*—Preparation for mathematical Olympiads like Regional Mathematical Olympiad (RMO) would be hard without any form of structured training, personal tutoring, and learning resources. In this paper, we have designed an intelligent tutoring system using AI tutor for solving complex mathematics questions with retrieval augmented generation (RAG), multimodal inputs, symbolic validation, and personalized learning methods. The system makes use of math OCR to process handwritten inputs, coordinate requests through the FastAPI webserver, perform vectorized semantic search with PostgreSQL database with pgvector support, and finally employ large language models to hint, evaluate, and tutor students. Instead of providing direct answers, the tutor uses the Socratic method to guide students. Some features of the system include problem generation, personalized recommendations, step-by-step grading of solutions, and monitoring of the learning process. Accuracy achieved by the system is 86% in final answers, 81% in logical reasoning, 89% in grading, and 91% in math OCR on a selected sample of 120 questions from the Omni-MATH database. Positive feedback was received from the preliminary trials conducted for RMO aspirants.**

***Index Terms*— Mathematical Olympiad, AI Tutor, Retrieval-Augmented Generation, Symbolic Verification, Math OCR, Personalized Learning, Olympiad Preparation, Educational Technology**

I. Introduction

The use of artificial intelligence has helped broaden the availability of resources for students. However, getting ready for competitions such as the Regional Mathematical Olympiad (RMO) continues to be an issue. The subject of olympiad math necessitates knowledge of concepts, proof-writing, and creativity in problem-solving, skills that cannot be achieved without proper mentoring.

Over the years, there has been a steady increase in student enrollment in mathematics olympiads at school level. Unfortunately, high-quality mentoring programs are only available to a small percentage of students who can afford them. For the others, books, previous year question papers, websites, and video lectures are some of the resources that are available. All these offer material but not interactive mentoring. Existing artificial intelligence mentoring applications are designed for standard coursework and examination purposes.

The proposed work is about developing a next-gen AI-driven RMO Tutor that integrates Retrieval-Augmented Generation (RAG), multimodal inputs (handwritten math, text, voice), symbolic reasoning, personalized memory, and grading into one system. Students can either write or type their math problems, and the handwritten math is automatically translated into LaTeX code by an AI component that recognizes handwriting and math symbols. Then, the problem is analyzed by AI models to provide relevant problems, tips, and explanations. Unlike traditional math tutors, rather than giving a solution directly, our system gives students hints to find the right solution. Mathematical reasoning is verified using LLMs and embedding similarity analysis.

The primary objectives of this work are:

1) Development of an AI-powered tutor which includes RAG, multimodal interaction, and symbolic verification for RMO preparation.
2) Development of auto-marking functionality which assesses correctness not only in the final answer but also in steps used to get it.
3) Increased accessibility to Olympiad mathematics training with the help of custom mock tests, recommendations, and analytics.

II. Literature Survey

AI-enabled educational systems cater mostly to the traditional test-taking environment, and cannot cope with open-ended and proof-related problems in the Olympiads format. We overview relevant approaches in Section II.

Mathematical Optical Character Recognition and Assessment. Gurgurov and Morshnev [1] developed an approach based on Swin Transformer and GPT-2 for handwriting-to-LaTeX conversion. This system is solely aimed at transcription; no reasoning analysis is provided. [2] applied CLIP and OCR algorithms to perform assessment of image inputs submitted by students, showing that the method achieves lower error rates than traditional text inputs while still lacking reasoning assessment functionality.

Reasoning verification. Cobbe et al. [3] presented GSM8K and verified that models that verify improve the accuracy of reasoning. However, the dataset focuses on elementary problems with no handwritten inputs. On the other hand, [4] proved that process monitoring, such as PRM800K, is better than result supervision, achieving an accuracy of 78 percent on MATH, although it has high costs for annotation and is unproven in learning environments.

AI in mathematics education. Garcia et al. [5] found that undergraduate engineers use artificial intelligence math programs for function instead of skill acquisition. According to [7], LLMs perform well with procedural problem-solving yet have low reliability in complex proofing. [8] performed a review on the current state of handwritten-input ITS systems and observed that only touch-screen algebraic problems were covered for English-speaking individuals.

Benchmarking and tutoring. Gao et al. [6] proposed Omni-MATH (4,428 Olympiad-level questions spanning 33 categories), on which even state-of-the-art systems achieve accuracy of only 60%. According to [9], a GPT-3 based tutor helped raise the accuracy level by 15 percentile points using spaced repetition technique; however, the experiment was very limited and conducted on a single domain. A tutor for statistics using RAG architecture [10] managed to avoid hallucinations but was evaluated only with the involvement of instructors.

No existing system combines Olympiad-level reasoning, personalized tutoring, symbolic verification, handwritten input analysis, and step-wise guidance in one platform. This paper addresses that gap.

III. Research Objectives

The aim of this study is to develop, implement, and analyze an AI Tutor for olympiad mathematics that incorporates a RAG model, agentic memory, symbolic verification, and multi-modal input/output capabilities (handwritten, LaTeX, speech, and text). The tutor must provide grounded responses based on information retrieval to avoid hallucinations, symbolically verify its responses to ensure mathematically sound reasoning, and leverage personalized records of the student's past actions to speed up learning and correct common mistakes. These claims are substantiated via benchmark testing on olympiad datasets and user studies against traditional study approaches.

IV. System Architecture

The architecture is based on the model of modular services architecture using React for the frontend, FastAPI for the backend, AI modules for various services, third-party APIs such as OpenAI, Math OCR, and event management services, and PostgreSQL as the database with vectors. Environment variables and API access keys are stored separately from the code of the application. Figure 1 shows the full architecture.

A. Frontend

The frontend has been made using React, TypeScript, and Vite. User sessions are validated via a JWT authentication middleware before accessing anything else. Underneath, there is an API client that directs all requests towards the server over REST. Four interfaces are used by students in this application; chat interface for tutorial sessions, mock test interface, problem submission interface, and analytics interface.

B. Backend API Layer

The backend uses FastAPI for routing and service coordination. Representative endpoints include `/api/chat` (tutoring conversation), `/api/mock-tests` (exam generation), `/api/submit-solution` (solution grading), and `/api/analytics` (performance tracking). Each endpoint performs minimal computation and delegates to the AI services layer.

C. AI Services Layer

This layer contains the five core modules. The *OCR Service* converts handwritten or scanned math into LaTeX via an external math OCR engine. The *Test Generator* assembles mock Olympiad exams from the problem database, balancing difficulty and topic coverage. The *Recommendation Engine* runs semantic similarity searches over a student's performance history to suggest practice problems. The *Grading Service* compares student answers against reference solutions and returns structured feedback. The *Tutor RAG Module* retrieves relevant problems, hints, and explanations from the database and passes them alongside the student's query to a large language model for response generation.

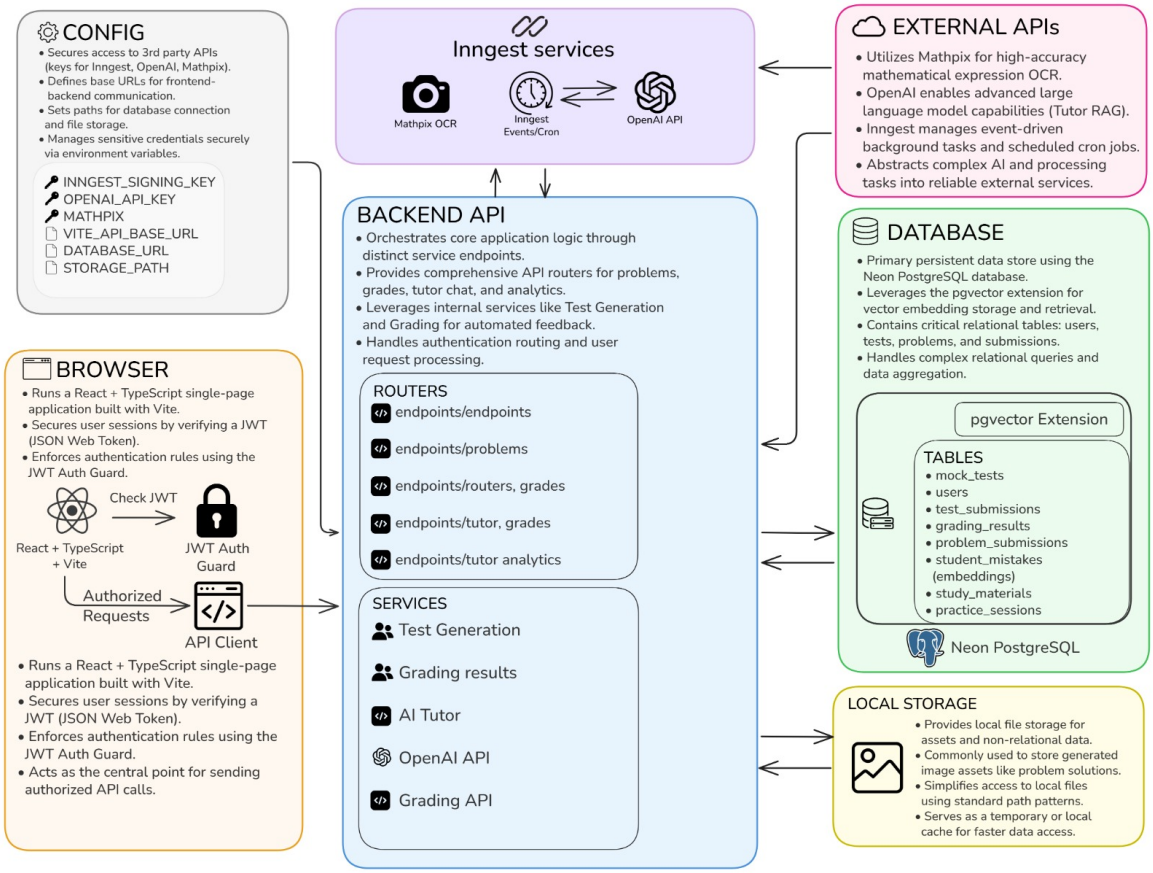

Fig. 1. End-to-end architecture of the proposed AI-powered Olympiad Tutor.

D. Data Persistence Layer

All structured data is stored in PostgreSQL, extended with pgvector for semantic search. Problem and mistake embeddings are generated using the `all-MiniLM-L6-v2` SentenceTransformer model, which produces 384-dimensional vectors. Tables cover users, mock tests, test and problem submissions, grading results, student mistakes (stored as vectors for similarity retrieval), study materials, and practice sessions. A local storage layer holds uploaded handwriting images and OCR-parsed outputs. Raw data is retained for auditing.

V. Dataset and Tool Description

A. Mathpix OCR

Mathpix OCR is a commercial STEM-focused OCR tool that processes images and PDFs containing printed or handwritten mathematics, tables, diagrams, and mixed layouts. It outputs LaTeX, AsciiMath, MathML, Markdown, or HTML, with confidence scores for each recognized element. The API exposes three endpoints: `v3/text` for single images, `v3/pdf` for full documents, and `v3/strokes` for digital ink. In our system, Mathpix converts uploaded handwritten math to LaTeX at the frontend; we evaluate both OCR conversion performance and downstream grading accuracy. OCR performance is measured using OCR Conversion Success Rate, while grading effectiveness is evaluated through grading consistency and feedback quality.

TABLE I
Omni-MATH Dataset Statistics

Parameter	Value
Total Problems	4,428
Domains	33
Difficulty Levels	10
Format	JSON
License	Apache 2.0

B. Omni-MATH Dataset

Omni-MATH [6] is an Olympiad-level benchmark containing 4,428 problems across 33 mathematical domains and 10 difficulty levels (Table I). Problems are drawn from the International Mathematical Olympiad (IMO), regional and national Olympiads, and other competition archives. Unlike simpler datasets such as GSM8K, Omni-MATH requires multi-step reasoning, formal logic, and proof intuition. We use it to evaluate step reasoning, solution verification, and error diagnosis.

VI. Automated Math Grading System

The grading system checks both the final answer and the validity of the solution steps. A student who follows

correct reasoning but makes a calculation error still receives partial credit.

A. Final Answer Matching

The system checks the student's final answer in three stages. First, both answers are normalized (whitespace, formatting noise, and unicode removed) and compared exactly. If they differ, an LLM judges whether they are mathematically equivalent (e.g., $\frac{1}{2}$ vs 0.5); the answer is accepted when the model's confidence is ≥ 0.85. If the LLM call fails, cosine similarity between answer embeddings is used as a fallback, again with a 0.85 threshold.

B. Logical Solution Evaluation

Even when the final answer is wrong, the student may have followed a valid approach. The system splits the solution into logical steps, and an LLM evaluates each step for mathematical validity, returning a logical score between 0 and 1, the total count of valid steps, the index of the first error, and a short error summary. If the LLM is unavailable, each student step is compared to the reference solution by semantic similarity; steps matching at ≥ 0.70 are counted as valid. This threshold serves as a heuristic fallback and does not guarantee mathematical or logical correctness.

C. Final Scoring Rule

The final score combines the correctness of the answer with the logical correctness of the steps:

$$\text{Final Score} = w_a \times \text{Answer Correctness} + w_s \times \text{Logical Score} \tag{1}$$

where w_a and w_s denote the answer weight and solution weight, respectively. These weights are assigned as follows:

$$(w_a,\, w_s) = \begin{cases} (0.4,\, 0.6), & \text{answer correct,} \\ (0.0,\, 1.0), & \text{answer wrong} \end{cases}$$

Finally, the percentage score is computed as:

$$\text{Percentage} = 100 \times \text{Final Score} \tag{2}$$

When the final answer is wrong, the score is determined entirely by the logical validity of the solution steps, allowing students to receive partial credit for valid reasoning despite an incorrect final result.

VII. System Flow

The overall flow of the system is as follows:

1) The student submits input via text, handwriting, or speech.
2) Preprocessing modules (OCR or STT) normalize the input.
3) The LLM/Agent Controller constructs a tutoring prompt using retrieval from the Vector DB, personalization from the Memory Layer, and pedagogy from the Hint Policy Engine.
4) Candidate solutions are verified by symbolic services for mathematical correctness.
5) The finalized response is delivered in text or voice through the student interface.
6) The Memory Layer updates student profiles with new attempts and progress summaries.
7) All data and artifacts are logged in the Persistence and Provenance Layer to ensure auditability.

VIII. Testing and Results

A. Testing Methodology

The system was tested at three stages. Initially, all backend services (Authentication, Problem Retrieval, OCR, Grading, Tutoring, Analytics) were tested for functional correctness using controlled input values. Subsequently, the accuracy of the proposed model was tested using a handpicked subset of 120 questions taken from the Omni-MATH dataset. Since the full-scale Omni-MATH benchmark exhibits a highly difficult performance curve, in which a typical frontier model, such as GPT-4o, only attains a base accuracy of 30.49%, testing our system against the entire collection of 4,428 questions would have been prohibitively expensive. In order to have a more balanced assessment, we chose a sample of 120 problems from each difficulty level using a stratified sampling method (selecting 40 problems at introductory levels T1-T2, 50 at intermediate levels T2-T3, and 30 at advanced Olympian levels T3-T4). Using this sample size made it possible for us to objectively test step reasoning, fallback efficiency, and error detection. The application was also posted to a relevant subreddit community and directly tested by a small sample size of students preparing for the RMO competition.

B. Evaluation Metrics

The following six metrics were used for evaluation of the proposed solution. Answer Accuracy is the ability of the system to determine an exact answer to a particular question. Logical Accuracy represents the accuracy of each logical step of the solution irrespective of the final solution obtained. Grading Consistency denotes how accurate the results provided by the automatic grader are compared to those provided in the reference solutions. OCR Success Rate denotes the efficiency of recognition and conversion of handwriting into math equations. User Engagement monitors user activity within the platform.

C. Results

1) Automated Evaluation Results: As seen in (Table II) above, the hybrid grading pipeline achieves impressive results in terms of its performance metrics on our

TABLE II
AUTOMATED EVALUATION RESULTS (COMPUTED ON N=120 SAMPLE SUBSET)

Metric	Value
Final Answer Accuracy	86%
Logical Step Accuracy	81%
Grading Consistency	89%
OCR Conversion Success Rate	91%
RAG Response Relevance	84%

TABLE III
REAL-WORLD TESTING RESULTS

Metric	Observation
Total Users	9
Mock Tests Attempted	15
Problem Submissions	90
Avg. Session Duration	35 minutes
Repeat Users	11%

evaluation subset selection. Such results are because of our customized prompts and embeddings used in the process and do not suggest that the model has achieved 86% accuracy in the full set of 4,428 problems, which is a far cry from the truth. **Observations:**

- LLM-based equivalence checking improved correctness detection
- Step-wise evaluation increased fairness in grading
- Minor errors occurred in combinatorics and proof-based problems

2) Real-World Testing Results: **Key Insights:**

- Users engaged most with mock tests and grading features
- Hint-based tutoring was preferred over direct answers
- OCR improved usability for handwritten inputs
- Recommendations showed moderate usage

3) User Feedback and Performance: The users found the hinting process, structured exercise design, handwriting recognition capability, and practice tests simulation to be useful. The most frequent issues encountered by the users related to wrong OCR recognition of the handwriting, lack of deep understanding in solving advanced tasks, and lack of geometry simulation capabilities. End-to-end API response took around 1.5 – 2.8 seconds. In the process of end-to-end API response, RAG retrieval took about 2.2 seconds, whereas grading operations took 2.5-4 seconds depending on the answer length.

D. Discussion

The hybrid pipeline for grading (LLM equivalence check and fall back to embeddings) explains the grade consistency at 89%. RAG retrieval helped ensure that tutoring responses remained aligned with Olympiad materials as opposed to being based on answer generation. Personalization proved less popular compared to mock tests; nevertheless, there were recurring users among the personalized recommendation category. Limitations include reduced accuracy on proof-based problems, the reliance on third-party APIs, the sensitivity of the OCR process to handwriting, and a small sample size for user tests.

While the use of semantic similarity provided additional robustness where LLMs failed to evaluate an answer, the method had some inherent flaws in answering math problems. Embeddings focus on the textual similarity of responses and are unable to detect the logical soundness of mathematical calculations and proof. As a result, incorrect responses can achieve very high similarity scores due to the text matching, which explains reduced performance in combinatorics and proof.

Symbolic verification and grading will be explored further in future work to address the limitations noted.

IX. CONCLUSION

This paper discusses an AI-enabled Olympiad Tutor, which helps enhance the accessibility and effectiveness of Regional Mathematical Olympiad training. This tutorial employs Retrieval-Augmented Generation, symbolic verification, handwritten OCR, personalized tutoring, and automatic marking techniques for efficient and educationally sound tutoring services.

Experiments confirmed that the tool performed well in terms of grading, OCR, and reasoning. The tool can effectively promote understanding using hints instead of answer generation.

Future work involves geometry visualization, better proof verification, greater scalability, and better adaptive learning systems.

REFERENCES

[1] D. Gurgurov, A. Morshnev. "Image-to-LaTeX converter for mathematical formulas and text," *arXiv preprint*, arXiv:2408.04015, 2024. [Online]. Available: https://arxiv.org/abs/2408.04015

[2] S. Baral, A. Botelho, A. Santhanam, A. Gurung, L. Cheng, N. Heffernan. "Auto-scoring student responses with images in mathematics," in *Proc. 16th Int. Conf. Educational Data Mining*, Bengaluru, 2023, pp. 362–369.

[3] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano *et al.* "Training verifiers to solve math word problems," *arXiv preprint*, arXiv:2110.14168, 2021. [Online]. Available: https://arxiv.org/abs/2110.14168

[4] H. Lightman, V. Kosaraju, Y. Burda, H. Edwards, B. Baker, T. Lee, J. Leike, J. Schulman, I. Sutskever, K. Cobbe. "Let's verify step by step," *arXiv preprint*, arXiv:2305.20050, 2023. [Online]. Available: https://arxiv.org/abs/2305.20050

[5] K. F. Garcia, A. K. S. Ong, M. J. J. Gumasing, C. R. V. Delos Reyes. "Engineering students' perceptions and actual use of AI-based math tools for solving mathematical problems," *Acta Psychologica*, vol. 256, p. 105004, 2025, doi: 10.1016/j.actpsy.2025.105004.

[6] B. Gao, F. Song, Z. Yang, Z. Cai, Y. Miao, Q. Dong, L. Li, C. Ma, L. Chen, R. Xu, Z. Tang, B. Wang, D. Zan, S. Quan, G. Zhang, L. Sha, Y. Zhang, X. Ren, T. Liu, B. Chang. "OMNI-MATH: A universal olympiad-level mathematic benchmark for large language models," in *Proc. 13th Int. Conf. Learning Representations (ICLR)*, 2025. [Online]. Available: https://arxiv.org/abs/2410.07985. Dataset: https://huggingface.co/datasets/KbsdJames/Omni-MATH

[7] I. Rizos, N. Gkrekas. "The impact of LLMs on mathematics education and research at the university," *Social Sciences & Humanities Open*, vol. 12, p. 101969, 2025, doi: 10.1016/j.ssaho.2025.101969.

[8] L. Rodrigues, F. D. Pereira, M. Marinho, V. Macario, I. I. Bittencourt, S. Isotani, D. Dermeval, R. Mello. "Mathematics intelligent tutoring systems with handwritten input: A scoping review," *Education and Information Technologies*, vol. 29, no. 9, pp. 11183–11209, 2024, doi: 10.1007/s10639-023-12245-y.

[9] R. Aebersold, M. Guggisberg, S. Konig. "Effective learning with a personal AI tutor: A case study," *Education and Information Technologies*, vol. 30, no. 2, pp. 1935–1957, 2025, doi: 10.1007/s10639-024-12888-5.

[10] S. R. Poudel, S. K. Sharma. "Designing a retrieval-augmented generative AI tutor for statistics education: Eliminating hallucinations and enhancing reliability," *Education and Information Technologies*, vol. 30, no. 3, pp. 3219–3243, 2025.