Final Proposal: Human-in-the-Loop Feedback-Based Math Agent

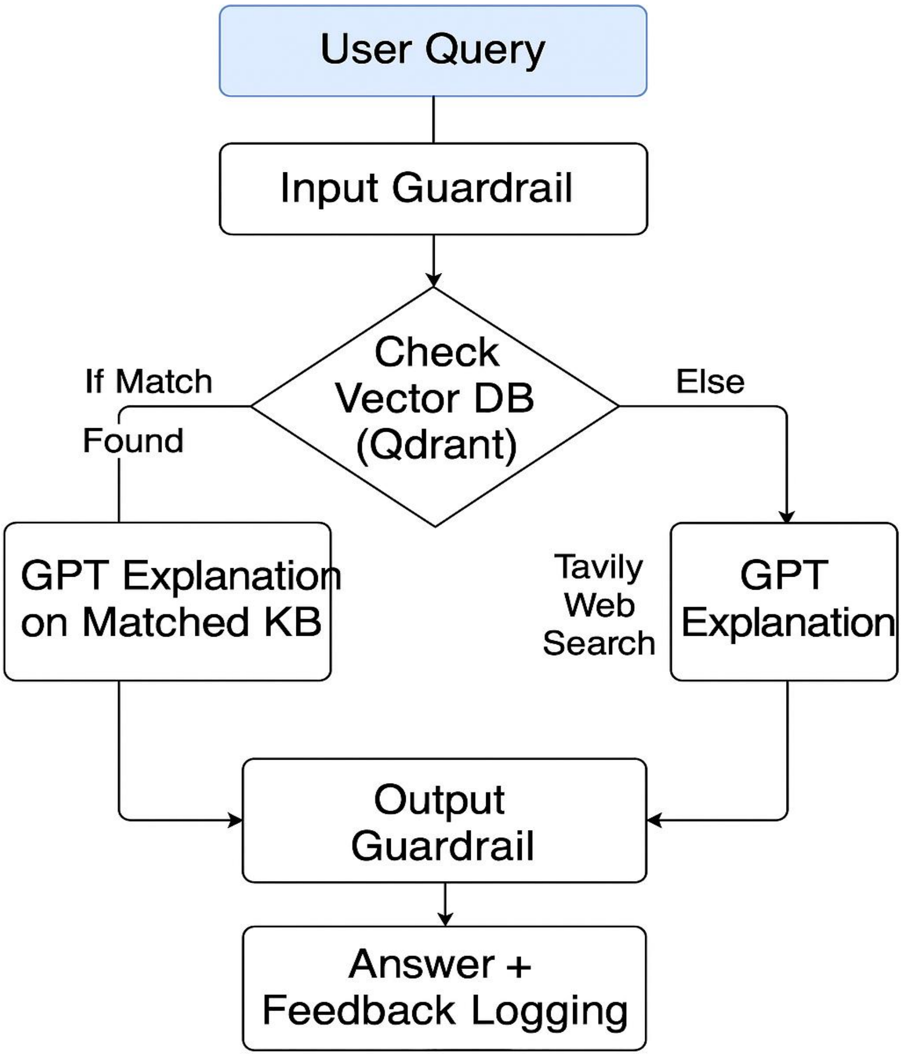
---

1. Project Overview

This project implements an **Agentic-RAG (Retrieval-Augmented Generation)** architecture to simulate a math professor capable of solving JEE-level math questions with step-by-step explanations. The agent intelligently routes the query through a knowledge base or web search, validates responses using guardrails, and incorporates human feedback for continuous improvement.

2. Architecture & Workflow

Flowchart Summary



Each query is evaluated through:

- **Input Guardrail**: Ensures math-only queries.
- **KB Check**: Uses Qdrant vector search with OpenAI Embedding.

- **Fallback**: Tavily API for web search when no KB match.
- **Explanation**: GPT-3.5 Turbo explains matched or searched content step-by-step.
- **Output Guardrail**: DSPy-based output filter for quality.
- **Feedback Loop**: UI lets users give thumbs up/down, stored in JSON logs.

3. Guardrails Description

To ensure that the Math Agent stays focused on educational math content, DSPy-based guardrails are implemented:

• **Input Guardrail**: Prevents the model from answering non-math or off-topic queries by rejecting them early.

• **Output Guardrail**: Validates that the generated explanation is safe, relevant, and free from hallucinations before returning it to the user.

This layered security ensures educational integrity and AI safety.

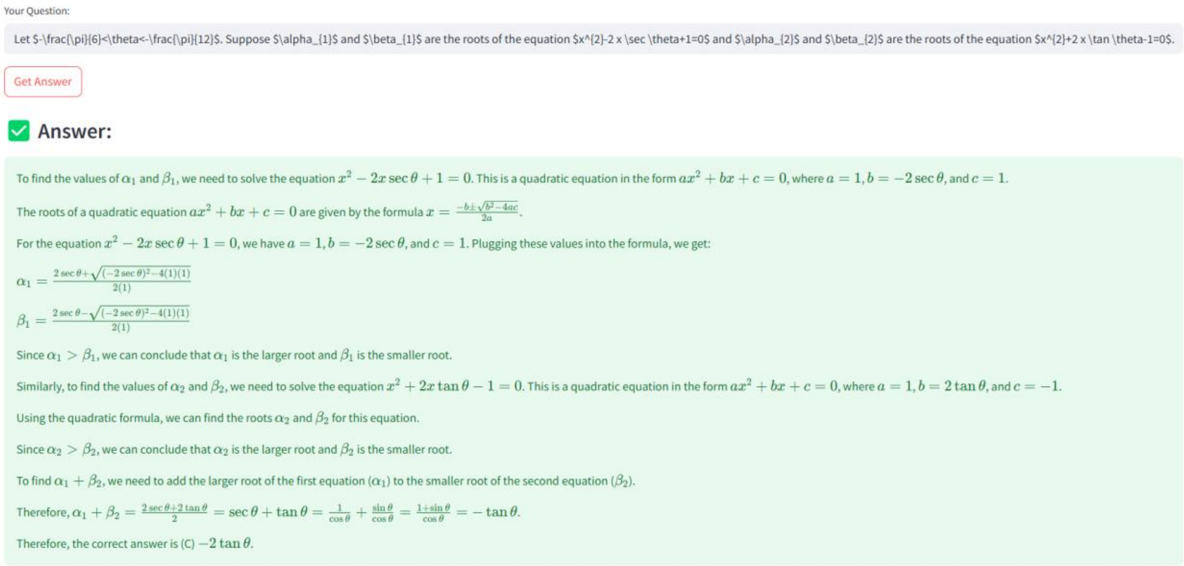
4. Knowledge Base Strategy

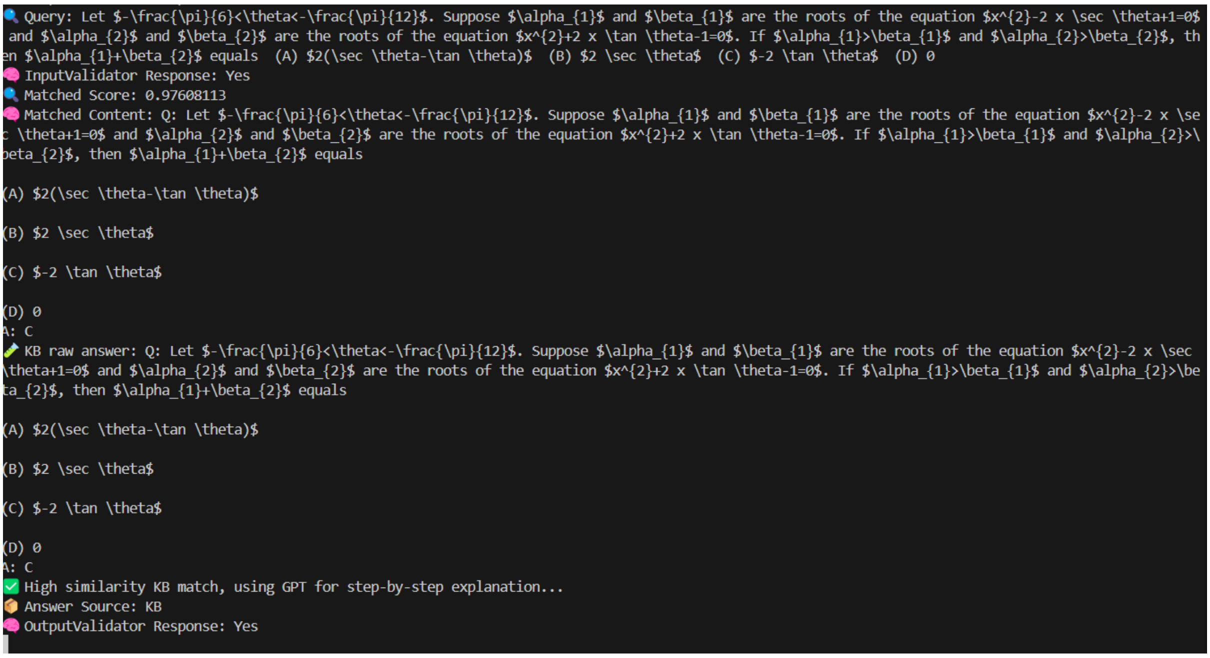
• **Dataset**: Filtered subset of JEEBench (math-only)

• **Format**: Each entry contains a math question and the correct answer (gold).

• **Embedding**: OpenAIEmbedding with Qdrant for fast similarity search.

• **Retrieval**: Top-1 similarity used to fetch relevant questions from KB.





```
Query: Let $-\frac{\pi}{6}<\theta<-\frac{\pi}{12}$. Suppose $\alpha_{1}$ and $\beta_{1}$ are the roots of the equation $x^{2}-2 x \sec \theta+1=0$
 and $\alpha_{2}$ and $\beta_{2}$ are the roots of the equation $x^{2}+2 x \tan \theta-1=0$. If $\alpha_{1}>\beta_{1}$ and $\alpha_{2}>\beta_{2}$, th
en $\alpha_{1}+\beta_{2}$ equals  (A) $2(\sec \theta-\tan \theta)$  (B) $2 \sec \theta$  (C) $-2 \tan \theta$  (D) 0
InputValidator Response: Yes
Matched Score: 0.97608113
Matched Content: Q: Let $-\frac{\pi}{6}<\theta<-\frac{\pi}{12}$. Suppose $\alpha_{1}$ and $\beta_{1}$ are the roots of the equation $x^{2}-2 x \se
c \theta+1=0$ and $\alpha_{2}$ and $\beta_{2}$ are the roots of the equation $x^{2}+2 x \tan \theta-1=0$. If $\alpha_{1}>\beta_{1}$ and $\alpha_{2}>\
beta_{2}$, then $\alpha_{1}+\beta_{2}$ equals

(A) $2(\sec \theta-\tan \theta)$

(B) $2 \sec \theta$

(C) $-2 \tan \theta$

(D) 0
A: C
KB raw answer: Q: Let $-\frac{\pi}{6}<\theta<-\frac{\pi}{12}$. Suppose $\alpha_{1}$ and $\beta_{1}$ are the roots of the equation $x^{2}-2 x \sec
\theta+1=0$ and $\alpha_{2}$ and $\beta_{2}$ are the roots of the equation $x^{2}+2 x \tan \theta-1=0$. If $\alpha_{1}>\beta_{1}$ and $\alpha_{2}>\be
ta_{2}$, then $\alpha_{1}+\beta_{2}$ equals

(A) $2(\sec \theta-\tan \theta)$

(B) $2 \sec \theta$

(C) $-2 \tan \theta$

(D) 0
A: C
High similarity KB match, using GPT for step-by-step explanation...
Answer Source: KB
OutputValidator Response: Yes
```

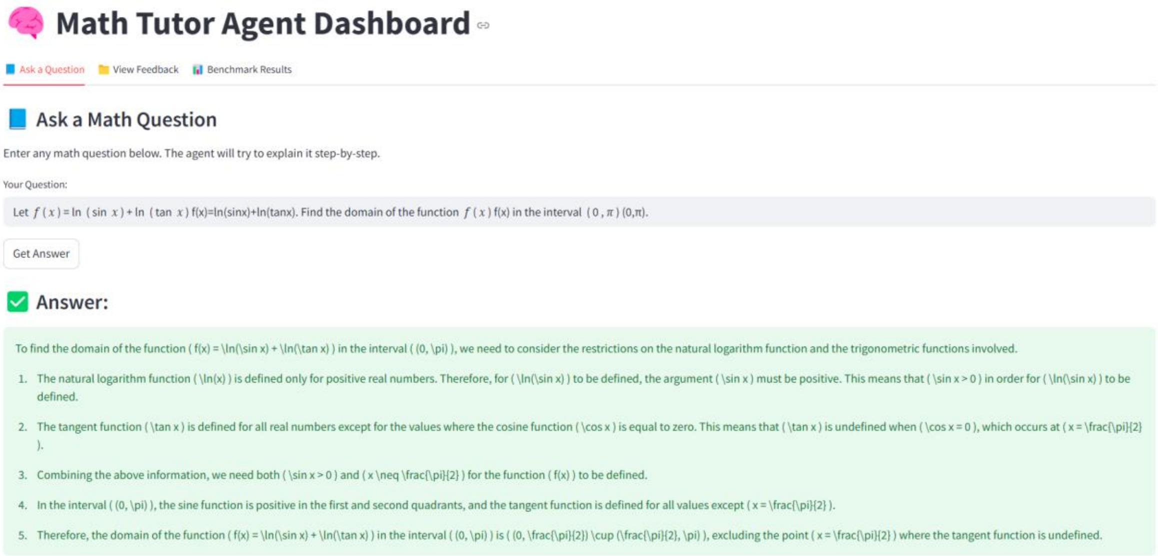
5. Web Search Fallback Strategy

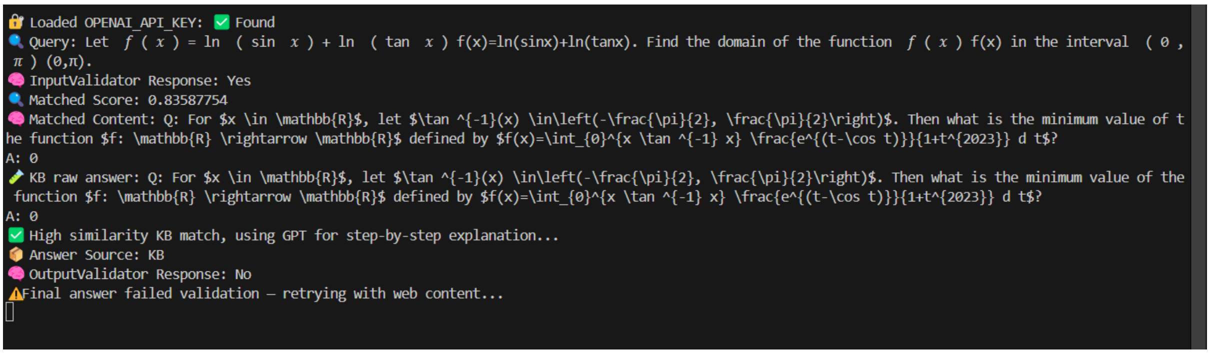
If no match is found in the vector DB or the similarity score is low, the agent automatically performs a web search using the Tavily API.

• **API**: Tavily (Basic Search)

• **Explanation**: GPT-3.5 uses search result to generate a valid answer.

• **Filtered by**: Output guardrail to maintain quality.





6. Human-in-the-Loop Feedback

Users can rate each answer using thumbs up/down in the Streamlit UI.

• Logged as JSON files in `logs/feedback_log.json`

• **Each log includes**: question, answer, feedback status

• Feedback is used to refine future outputs and assess agent quality

Sample Log Entry:

```
{
  "question": "Evaluate ∫sin(x)dx",
  "answer": "The integral is -cos(x) + C",
  "feedback": "positive"
}
```

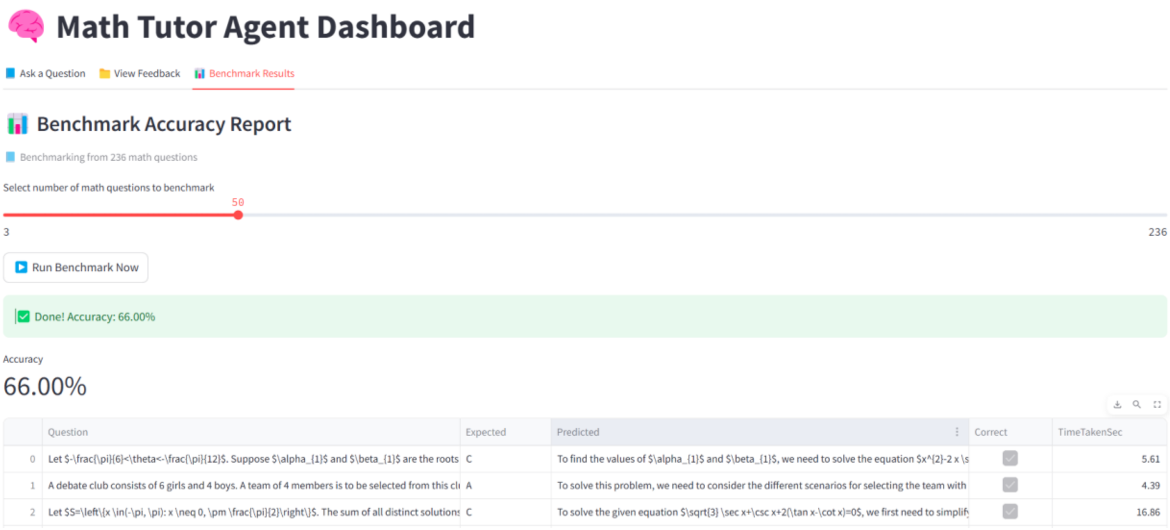
7. Benchmarking on JEEBench

• **Dataset**: 236 math questions from JEEBench

• **Method**: Custom benchmarking tool in Streamlit

• **Strategy**: Compare predicted answer with gold label

• **Accuracy**: 66% (33 correct out of 50 questions)



8. Deliverables

• PDF Proposal

• Streamlit App Source Code

• `rag/query_router.py`, `app/streamlit.py`, `app/benchmark.py`

• Feedback Logs

• Benchmark Results

• Demo Video showcasing:

  - Agentic routing

  - Feedback logging

  - Benchmarking