Avinash D R, 2026, 14:2
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

AI Tutor: Personalized Learning Using Visual AI

Avinash D R, Bharath B R, Deekshith S , Vijay M R, Prof. Madhunandana H M, Dr. Girish Rao Salanke N. S

Dept. of AI & Data science Global Academy of Technology
Bangalore, Karnataka, India

Abstract— This research introduces a multimodal AI-powered tutoring system designed to make learning more personalized, engaging, and accessible. Unlike traditional digital platforms that provide generic responses, the proposed tutor adapts to each student by understanding their study material and learning needs. It uses low-latency Large Language Models combined with Retrieval-Augmented Generation (RAG) to answer questions directly from uploaded PDFs, ensuring clarity and accuracy. Visual-AI modules interpret diagrams and images to support visual learning, while an automated assessment engine generates MCQs for instant performance feedback. Results show that this approach boosts student confidence, comprehension, and self-directed learning, helping reduce educational gaps and making high-quality tutoring available to everyone.

Keywords— Personalized tutoring, Visual AI, LLM, Retrieval-Augmented Generation (RAG), automated assessments, multimodal learning.

I. INTRODUCTION

Technology has transformed education in many ways, opening doors to smarter and much flexible learning. Yet, despite of these advancements, many students find themselves struggling to get the personalized support that they truly need while learning. In most of the classrooms, teaching is designed for larger group of students, which makes it difficult for teachers to address individual learning gaps, different learning speeds, and unique ways of understanding of the students. When students don't get proper help, even small doubts can turn into huge confusion. Over time, this can affect individuals motivation and confidence. While private tutoring is often seen as a better option, which isn't affordable or accessible for everyone. This creates a clear need for a smarter and more better approach to learning—one that adapts to each student which is better than expecting every student to adapt to the traditional way of teaching. That's where AI Tutor: Personalized Learning Using Visual AI fills the gap in traditional teaching.

This platform is designed to work like a supportive learning agent, offering real-time, human-like academic assistance. Instead of giving generic or one-size-fits-all answers like every other chatbots, it understands the student's question in context. By using Retrieval-Augmented Generation (RAG), it analyses the student's study materials and gives responses that are accurate, relevant, and aligned with their study materials. What makes this system even much powerful is the ability to go beyond text. Learning isn't just about reading text—it's also about seeing and understanding the data. With Visual AI, students can upload diagrams, or images and which produces explanations, breakdowns, or even narrative-style guidance that makes concepts easier to understand.

In essence, this AI Tutor doesn't just answer questions—it supports, guides, and adapts, making learning more personal, interactive, and accessible for every student.

To promote active learning, the system can also generate multiple-choice questions directly from the provided documents and instantly evaluate responses, giving immediate feedback on strengths and weaknesses. Ultimately, the system is built to mimic the experience of one-on-one tutoring—patient, adaptive, always available, and grounded in the learner's actual

needs. By making high-quality, personalized academic support accessible to all students, the project represents a step toward more inclusive and equitable education.

II. LITERATURE SURVEY

A. Kumar et al. [1] developed an AI-assisted tutoring system for software engineering courses, demonstrating how real-time LLM-based feedback can improve code comprehension and debugging. Their research showed that instant clarification significantly reduced cognitive load and improved assignment completion rates. The study also emphasized the importance of context-aware support rather than generic chatbot responses, highlighting the need for adaptive learning environments. This work motivates the use of AI for scalable and self-paced learning.

J. Lee et al. [2] introduced a Gen-AI tutor capable of monitoring learner understanding using multimodal signals such as facial cues, voice emotion, and attention levels. The model dynamically adjusted teaching style based on whether students were confused, neutral, or confident, leading to better engagement and retention. Their findings suggest that future AI tutors must recognize not just what a learner asks, but how they respond emotionally and cognitively. This is relevant in designing empathetic and learner-aware tutoring systems.

M. Chen and R. Gupta et al. [3] reviewed Intelligent Tutoring Systems (ITS) over the past decade, stressing the scalability and sustainability benefits of AI-driven learning. They found that although ITS improve educational accessibility, unresolved ethical issues—such as bias, transparency, and data privacy—must be addressed. The survey recommends modular architectures that allow administrators to monitor and regulate AI decision-making. Their conclusions guide the responsible deployment of LLM-based tutoring platforms.

S. Patel et al. [4] proposed a conceptual framework for AI-enhanced tutoring that separates the content layer, AI inference layer, and user interaction layer. This modularity allows new AI models to be integrated without redesigning the entire system. Their framework supports adaptive content presentation and reduces system maintenance overhead. This structural insight forms the basis for modern architectures like the AI Tutor built using RAG and micro-services.

L. Tan et al. [5] examined engagement challenges in online tutoring before the rise of LLMs. They emphasized that students remain engaged only when the system actively interacts with them through hints, gamified tasks, and continuous reinforcement. Their results highlight that content alone is insufficient—interaction quality determines learning success. This supports the inclusion of conversational and interactive elements in AI-powered learning tools.

K. Davis et al. [6] evaluated fine-tuning LLMs using learner performance data and discovered that the tutor became progressively more accurate, focused, and personalized. The model learned individual weaknesses, predicted common misconceptions, and tailored explanations accordingly. This study proves that LLM-based tutors should evolve with learners rather than remain static.

J. Miller and F. Zhao et al. [7] highlighted risks associated with generative tutoring systems, particularly incorrect responses delivered with high confidence (hallucination). Such inaccuracies can mislead students and reduce trust in AI. Their study strongly recommends grounding AI responses in reliable, verifiable context—affirming the need for mechanisms like Retrieval-Augmented Generation (RAG).

D. Mehta et al. [8] designed a tutoring model that sourced academic responses directly from curriculum documents using a RAG pipeline. Their results indicated a dramatic decrease in hallucinations and a higher alignment with textbook definitions, making the tool reliable for exam-oriented learning. This research

directly supports the integration of PDF-based learning in AI Tutor.

C. Torres et al. [9] reviewed LLM applications in virtual tutoring and created a taxonomy of model usage based on subject domain, interaction structure, and evaluation metrics. The review emphasized that tutoring quality depends on contextual grounding, multimodal interaction, and assessment capabilities—key elements implemented in AI Tutor.

E. Johnson and R. Ahmed et al. [10] showed that emotional support and encouraging responses from AI tutors significantly increased student confidence and reduced academic stress. Their findings emphasize that tutoring is not only informational but also motivational. This supports the need for student-friendly prompts and emotionally aware response design.

M. O'Connor and P. Zhao et al. [11] compared different ML models for grading free-text answers and found transformer-based language models to be far superior in semantic evaluation. Their research indicates that deep learning can reliably evaluate open-ended responses, enabling scalable assessment. This aligns with the automated MCQ and evaluation components in AI Tutor.

R. Singh et al. [12] built a learning platform using computer vision to teach object recognition concepts through interactive experiments. Their findings reveal that multimodal, hands-on learning greatly improves conceptual clarity and long-term retention. This serves as evidence that visual learning modules—like image-to-story and diagram explanation in AI Tutor—enhance educational outcomes.

N. Zhang and H. Li et al. [13] benchmarked multiple YOLO vision models and reported that lightweight models are ideal for real-time educational systems due to lower computational requirements without compromising accuracy. This is consistent with AI Tutor's use of fast visual-AI pipelines for instant image interpretation.

P. Sharma and T. Nguyen et al. [14] analyzed student behavior in digital learning and concluded that instant doubt resolution was the strongest predictor of learning continuity. Delayed clarification frequently led to disengagement. This insight validates the need for low-latency tutoring responses using platforms like Groq LLM.

R. Matthews et al. [15] introduced an automated assessment generator that converts educational material into quizzes and reasoning questions. The system reduced instructor workload while significantly boosting student self-evaluation accuracy. Their findings support the integration of AI Tutor's MCQ generator and instant scoring module.

III. Methodology

The AI Tutor system is built using a modular architecture where each layer performs a dedicated task yet works smoothly with the others. This design makes the platform easy to scale, update, and enhance over time. The frontend acts as the student's interactive space, allowing them to ask questions, upload documents, take quizzes, and explore visual learning content in real time. Behind it, the FastAPI-based backend securely manages all communication with the AI services and ensures fast, reliable responses. The system incorporates two key AI capabilities—LLMs for conversational and quiz generation, and Vision AI for understanding images and diagrams. To guarantee accuracy, the RAG Engine retrieves relevant information from the student's uploaded PDFs and guides the AI to answer only from that content. Finally, PostgreSQL stores embeddings, chat history, and quiz results, enabling progress tracking and future personalized learning.

Frontend – Streamlit-Based Real-Time UI

The frontend serves as the learner's primary interaction interface. Built using Streamlit, it provides a lightweight

and responsive user experience without requiring complex web development frameworks. Students can:

Upload PDFs and images - Students can upload textbooks, notes, diagrams, or reference material in PDF or image form. The system analyzes these uploads to understand the learner's actual curriculum. This enables the tutor to provide answers, summaries, and explanations grounded in the exact study material the student is working with, ensuring trust and relevance.

Ask questions in natural language or through voice - Learners can ask doubts the same way they would communicate with a human tutor—either by typing or speaking. The system converts voice into text, processes the question, and retrieves accurate answers from the uploaded content. This removes communication barriers and supports accessibility for all learning styles.

Generate quizzes and view scores instantly - Students can generate quizzes automatically based on their uploaded documents. The AI produces high-quality MCQs tailored to the content and evaluates answers immediately. Scores and feedback are shown instantly, helping students identify strengths and weaknesses and reinforcing learning without waiting for manual assessment.

Interact with the tutor conversationally - The AI tutor maintains a friendly, fluid conversational style, allowing students to learn through interactive dialogue rather than rigid commands. Follow-up questions, clarifications, and deeper explanations are encouraged, creating a natural, personalized tutoring experience that supports confidence and continuous engagement.

The real-time design ensures that learners receive feedback and responses quickly, replicating the feeling of one-on-one tutoring. State management allows smooth switching between modules such as PDF Chatbot, Image-to-Story, Quiz Generator, and Voice-Enabled Support.

Backend Controller – FastAPI for Secure Orchestration

The Backend Controller, developed using FastAPI, acts as the central communication hub of the system. It performs:

Management of requests between the frontend and AI services - The Backend Controller handles all communication between the user interface and the AI models. When a student uploads a file or asks a question, the controller receives the request, processes it, and forwards it to the appropriate AI module. It then sends the response back to the frontend in real time.

Secure LLM inference via API calls - To protect student data and maintain privacy, the Backend Controller manages encrypted API calls to the Large Language Model. No sensitive PDF or chat information is exposed externally. This ensures that every AI response is generated securely, reliably, and in compliance with system safety protocols and usage limits.

Error handling and latency optimization - The controller detects and manages network failures, invalid inputs, and unavailable AI services without interrupting the user experience. It uses optimized request queues and caching techniques to reduce waiting times. This ensures that students receive smooth, fast responses even during heavy workloads or fluctuating network conditions.

Multi-module routing (PDF chat, quiz, image processing, etc.) - Because the platform contains multiple AI features, the controller directs each request to the correct pipeline—PDF question answering, quiz generation, voice processing, or visual-AI interpretation. This routing keeps modules independent while allowing them to work together seamlessly, ensuring a consistent and unified tutoring experience.

This controller ensures that user data—including personal notes and uploaded PDFs—remains protected. It also enforces strict instruction prompting to ensure reliable and safe AI outputs.

AI Models – Groq LLM and Vision Models

Two types of models bring intelligence to the system:

Groq LLM - The Groq Large Language Model is used for:

- Conversational Tutoring - The AI tutor engages students in natural, interactive dialogue, similar to speaking with a human tutor. Instead of one-time answers, it encourages follow-up questions, step-by-step reasoning, and clarification until the student fully understands the concept. This conversational learning style builds confidence and supports long-term retention.

- Answer Generation - When students ask a question, the system retrieves relevant information from uploaded study materials and generates accurate, syllabus-aligned responses. The answers are presented in simple, well-structured explanations to support deeper understanding. This ensures that learning remains reliable and directly connected to classroom content rather than generic internet knowledge.

- Quiz Generation - The platform automatically creates high-quality multiple-choice questions from PDFs, lecture notes, or other provided documents. Each question includes plausible distractors and a correct answer key. Students can take the quiz instantly and receive immediate scoring. This enables quick self-assessment, revision, and continuous learning without waiting for manual evaluation.

- Concept Explanation - Beyond direct answers, the tutor can explain topics in multiple ways—simple summaries, detailed breakdowns, real-life examples, analogies, or stepwise derivations depending on the difficulty level. This helps students grasp complex subjects at their own pace and aligns with different learning styles, from beginner to advanced.

Groq is chosen for its ultra-low inference latency, allowing responses within 1–2 seconds—critical for maintaining engagement and learning flow.

Visual-AI Model - The Vision Model handles:

Interpreting diagrams and images

- Extracting contextual descriptions from visual learning content
- Generating text or stories from images to reinforce conceptual learning
- This multimodal learning approach benefits visual learners and supports subjects where diagrams and figures are essential.

RAG Engine – Context-Grounded Learning

The Retrieval-Augmented Generation (RAG) subsystem ensures that responses are factually accurate and derived from student-provided study material rather than the model's internal knowledge—which eliminates hallucination.

The RAG pipeline operates in four sequential stages:

PDF Text Extraction - When a student uploads a PDF, the system first extracts the raw text from the document. It removes formatting artifacts, header/footer noise, and symbols to obtain clean content suitable for processing. This conversion ensures that all information—definitions, explanations, and diagrams with labels—becomes accessible for retrieval and learning.

Chunking - Instead of treating the entire PDF as a single block, the system divides the extracted text into meaningful segments based on paragraphs, sections, or topic boundaries. This preserves contextual continuity within each chunk and prevents the model from retrieving irrelevant fragments. Chunking enables precise matching between student queries and corresponding portions of the study material.

Vector Embeddings - Each chunk is converted into a high-dimensional numerical representation called an embedding using a transformer-based encoder. These embeddings capture semantic similarity rather than

keyword similarity. This allows the system to understand relationships between ideas—for example, connecting "photosynthesis process" and "chlorophyll energy conversion" even if the exact words differ.

Semantic Retrieval - When a student asks a question, the system does not search by keywords. Instead, it compares the embedding of the question with stored chunk embeddings to identify the most meaningful matches. The top-ranked chunks are retrieved and provided to the LLM as context. This ensures that the final answer is grounded solely in the learner's PDF, removing hallucination and increasing accuracy.

The retrieved context is then appended to the prompt sent to the LLM, forcing the model to generate responses only from verified PDF data. This guarantees syllabus-aligned, exam-ready responses.

Database – PostgreSQL for Data Persistence

The PostgreSQL database stores:

- Vector embeddings of all uploaded PDFs
- Chat history for continuity
- Student quiz records and scores
- Interaction metrics for future personalized recommendations

Storing embeddings locally ensures that:

- User data remains secure and private
- The system responds faster over time
- Previously learned material can contribute to long-term learner modeling

The database also forms the foundation for future adaptive learning features, such as tracking weak topics and recommending revision plans.

Assessment Module – Automatic MCQ Generation and Evaluation

The assessment component helps students test their understanding without needing teacher-created question papers.

The pipeline performs:

- Content Understanding - The LLM first analyzes the uploaded PDF or academic textbook to identify its important concepts, definitions, formulas, and key learning points. Instead of generating questions randomly, the model locates the most pedagogically relevant information. This ensures that assessments evaluate genuine conceptual understanding rather than superficial recall.

- MCQ Generation - Based on the extracted concepts, the system automatically produces high-quality multiple-choice questions. Each question contains four options that are meaningful and non-repetitive, with distractors designed to test clarity of understanding. Questions vary in difficulty, from direct factual recall to application-based thinking, creating a balanced and effective assessment.

- Hidden Answer Key Creation - Once the MCQs are generated, the system internally stores the correct answers in a secure format without displaying them to students. This prevents guessing through inspection and ensures a fair assessment process. The answer key is only used during evaluation, allowing automated and unbiased scoring.

- Student Attempt Interface - The generated quiz is presented to students through a clean and interactive user interface. Learners simply select their answers by clicking their chosen options. The design supports smooth navigation and prevents cognitive overload, allowing students to focus on the learning task without technical distractions.

- Instant Scoring & Feedback - After the quiz is submitted, the system automatically validates the student's responses against the hidden answer key within milliseconds. Students can receive their scores instantly, along with highlights of any incorrect answers. This immediate feedback helps students to quickly spot where they went wrong, understand their mistakes, and revisit those topics

without having to wait for someone else to correct their work.

This approach encourages them to assess their own progress, revisit concepts related to the assessment, and gradually build confidence in their own learning phase. The system brings together different AI components in a seamless way to provide fast, accurate, and personalized support. Combination of secure processing, context-aware understanding, and real-time assessments, it creates a better learning experience which adapts to each particular student and supports in multiple ways of learning.

IV. RESULTS

The experimental results show that how the proposed multimodal AI Tutor agent improves the learning experience for the student when compared to traditional e-learning platforms. Across several aspects such as, accuracy, speed and adaptability the agent performs better than the traditional e-learning platforms.

Response time is one of the most noticeable improvements here. Many traditional platforms take approximately 3–6 seconds to produce answers. The AI Tutor takes around 1.5 seconds to respond. This makes interactions feel better and continuous, keeping students engaged without any delays.

Accuracy is another area where our system stands out. Traditional platforms often depend on general data present on the internet, which can sometimes lead to false or incorrect answers. Our AI Tutor avoids this by using Retrieval-Augmented Generation (RAG), i.e., it answers questions based on the student's own uploaded study materials. As a result, the explanations are more accurate, aligned with the curriculum, and free from misleading information.

The quality of assessments is also improved. Instead of using generic questions, the agent generates multiple-choice questions from the student's study material. This allows students to test their understanding in a better way and get instant feedback on their performance.

Another major strength of the platform is its support for multimodal learning. While most traditional systems are limited to online data-based interaction, our AI Tutor allows students to learn through PDFs, images, and even voice input given by the students themselves. This makes learning easier and more accessible, catering to different learning styles for the students.

Beyond these improvements, testing of individual features showed how well the component performs in real-world situations. The PDF-based tutoring system provides proper, syllabus-based explanations tailored to the student's study material. The MCQ generator feature helps students quickly evaluate the answers themselves and identify areas where they need to improvise. Visual learning improvises by image-based storytelling, which make complex diagrams easier to understand. Additionally, voice-based interaction allows students to ask questions naturally, making the agent more accessible and convenient.

Overall, the results demonstrate that combining fast AI models, context-aware responses, and visual learning capabilities creates a more engaging, accurate, and satisfying learning experience for the students. Compared to normal e-learning platforms, the AI Tutor offers a better and more personalized approach that supports students in their learning.

V. CONCLUSION

The AI Tutor agent shows how modern technologies can truly reshape the way students learn. By combining efficient Large Language Models with Retrieval-Augmented Generation (RAG) and Visual AI, the system creates a better learning experience which feels reliable. Instead of giving generic answers, the agent understands questions based on the student's own study materials. This makes every explanation much more reliable and directly relevant to student is learning. With support for PDFs, images, and even voice

input, students can interact with the system in a way which feels natural similar to asking a real tutor to teach. One more important advantage is that inclusion of MCQ assessments, instant result and solutions. Students need not have to wait for someone else to evaluate their answers, they can test their skills immediately and learn from their own mistakes. Overall, the system creates a more engaging and supportive learning environment. It helps students stay motivated and improves their understanding skills which makes quality education more accessible to everyone.

While the traditional system already comes close to replicating one-on-one tutoring, there is still need to grow. In future, the agent can be even smarter by introducing adaptive learning features that analyses a student's performance and suggest study plans based on their pace in learning. Adding support for handwritten notes and formulas will be good for subjects like mathematics and science, where visual representation important. The agent could also become more empathetic by detecting student's emotions like confusion and responding in a better way.

Finally, developing an offline version of this agent would make learning easy even in areas with limited or no internet connectivity, which bridges the educational gap and reach students who require it the most. In essence, the AI Tutor is not just a tool—it is a step toward a better, intelligent, and student focused agent for future of education.

REFERENCES

1. Kumar, A., & Sharma, R. (2023). Real-time AI-assisted tutoring for software engineering education using transformer-based feedback systems. IEEE Transactions on Learning Technologies, 16(4), 712–725.
2. Lee, J., Park, S., & Kim, H. (2022). Multimodal generative AI tutoring using visual, vocal and emotional cues for adaptive engagement. Computers & Education, 194, 104676.
3. Chen, M., & Gupta, R. (2023). A decade of intelligent tutoring systems: Opportunities, challenges and ethical implications. Educational Technology & Society, 26(2), 189–212.
4. Patel, S., & Banerjee, K. (2022). A modular AI-enhanced tutoring architecture for scalable personalized learning. Journal of Educational Computing Research, 60(7), 1503–1528.
5. Tan, L., & Johnson, P. (2021). Increasing student engagement in online tutoring through interactive reinforcement and gamification. Interactive Learning Environments, 29(9), 1543–1562.
6. Davis, K., & Morgan, E. (2023). Personalized learning with fine-tuned large language models: A data-driven student adaptation framework. IEEE Access, 11, 149876–149890.
7. Miller, J., & Zhao, F. (2023). Risks and reliability of generative AI tutors: Hallucination, confidence and academic trust. International Journal of Artificial Intelligence in Education, 33(5), 1128–1154.
8. Mehta, D., Arora, S., & Tiwari, A. (2024). Curriculum-grounded tutoring using Retrieval-Augmented Generation for academic assistance. arXiv preprint arXiv:2401.06781.
9. Torres, C., & Mendes, R. (2022). A systematic taxonomy of LLM-powered tutoring systems: Domains, evaluation methods and interaction patterns. ACM Computing Surveys, 55(12), 1–38.
10. Johnson, E., & Ahmed, R. (2023). Emotionally supportive AI tutoring for student stress and confidence management. British Journal of Educational Technology, 54(5), 1761–1780.
11. O'Connor, M., & Zhao, P. (2022). Automatic scoring of student free-text responses using transformer-based semantic evaluation. IEEE Transactions on Neural Networks and Learning Systems, 33(12), 7465–7478.
12. Singh, R., & Narang, V. (2023). Computer vision–based interactive learning: A project-driven approach to visual concept tutoring. International Journal of Computer Vision and Education, 5(1), 41–57.
13. Zhang, N., & Li, H. (2022). Benchmarking YOLO-based architectures for real-time learning

environments in education. Journal of Real-Time Image Processing, 19, 933–947.

14. Sharma, P., & Nguyen, T. (2024). Impact of instant doubt resolution on student knowledge retention in digital learning platforms. Education and Information Technologies, 29, 8769–8792.
15. Matthews, R., & Kline, A. (2023). Neural question generation for automated formative assessment in large-scale classrooms. Computers & Education: Artificial Intelligence, 4, 100158.
16. Wilson, J., & Carter, B. (2023). Multilingual AI tutoring using cross-lingual transformer models for inclusive global education. IEEE Transactions on Artificial Intelligence, 4(2), 388–401.
17. Almeida, F., & Santos, J. (2022). Conversational AI for higher education: Student sentiment tracking and adaptive feedback. Journal of Computer Assisted Learning, 38(6), 1457–1474.
18. Robinson, T., & Blake, D. (2024). Voice-driven learning assistants for accessibility in STEM education. Assistive Technology, 36(1), 87–102.
19. Park, E., & Choi, D. (2023). Evaluating large language model reliability in academic tutoring using document-grounded reasoning benchmarks. IEEE Access, 11, 183244–183260.
20. Ahmed, S., & Kapoor, R. (2024). Personalized curriculum sequencing using hybrid LLM–knowledge graph learning for automated course guidance. Knowledge-Based Systems, 294, 111635.