IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 17, April 2026
Impact Factor: 8.2

AI-Powered Virtual Tutor for Realtime Student Doubt Resolution

Suhasini A[1], Aswini N[2], Bhavadharani VPV[3], Dhivya S[4], Divya[5] R, Subasri S[6]
Professor, Department of Computer Science and Engineering.
Final Year students, Department of Computer Science and Engineering.
Annamalai University, Chidambaram, Tamil Nadu, India.

Abstract: *Artificial Intelligence is playing a transformative role in education by enabling interactive, personalized, and always-available learning support. This work presents an AI-powered virtual tutor designed for real-time student doubt resolution in core domains such as Artificial Intelligence, Machine Learning, Deep Learning, Natural Language Processing, Neural Networks, and Data Science. The proposed system supports multimodal interaction through text queries, voice-based input, PDF-assisted questioning, and image-based understanding. A Flask-based backend coordinates user requests, retrieves prior session context from SQLite, and communicates with Groq-hosted large language models to generate structured tutor-style explanations. The system further enhances learning through browser based speech synthesis, topic extraction, progress tracking, and quiz generation. Unlike static learning resources, the virtual tutor provides contextual, step-by-step, and encouraging responses that resemble human tutoring behavior. The overall architecture demonstrates how pre-trained foundation models, prompt engineering, lightweight web technologies, and persistent session memory can be combined into a practical educational platform without requiring custom model training infrastructure.*

Keywords: AI Tutor, Intelligent Tutoring System, Large Language Model, Prompt Engineering, Multimodal Learning, Student Doubt Resolution

I. INTRODUCTION

The rapid growth of Artificial Intelligence has significantly influenced the education sector by enabling systems that support adaptive, interactive, and personalized learning. In conventional academic environments, students depend heavily on faculty guidance, textbooks, and scheduled classroom sessions to clarify difficult concepts. However, such support is not always avai leverages lable at the exact time a learner needs help. As a result, students often face delays in clearing doubts, discontinuity in learning, and difficulty understanding technically complex topics at their own pace.The proposed AI-powered virtual tutor addresses this challenge by providing an educational assistant dedicated to AI-related domains. The system is designed as a web-based platform that allows students to ask questions in natural language, upload supporting files, interact through voice, and receive guided explanations in a conversational tutor-like style. Unlike a generic chatbot, the system is restricted to academic topics such as Artificial Intelligence, Machine Learning, Deep Learning, Natural Language Processing, Neural Networks, and Data Science.The significance of the project lies in its integration of multiple learning-support features within a single application. Students can type or speak a doubt, upload a PDF or image, receive contextual explanations, listen to spoken responses, track covered topics, and generate quizzes for self-assessment. The system combines frontend interaction, backend orchestration, prompt-controlled tutoring behavior, model inference, and persistent session storage into one practical educational workflow.This work also demonstrates a realistic engineering approach to educational AI. Rather than training a custom deep learning model from scratch, the application strong pre-trained large language models through API access. This reduces development overhead while still enabling advanced reasoning, multimodal understanding, and personalized explanation. Therefore, the project can be viewed as both an academic software system and a practical example of applying modern AI technologies to student-centered learning

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 17, April 2026
Impact Factor: 8.2

II. LITERATURE SURVEY

Recent studies show that AI-driven tutoring systems are becoming important in digital education because they can provide immediate responses, personalized explanations, and flexible access to learning resources. Dr. Pushpa D., Arpitha H. M., and Ananya M. (2025) proposed an AI-powered virtual tutor using generative AI to improve interactive learning support. Their work highlighted the ability of modern language models to explain concepts conversationally and reduce dependence on traditional static resources. Kestin, Miller, Klales, Milbourne, and Ponti (2025) reported that AI tutoring systems can outperform conventional in-class active learning under controlled conditions. Their results emphasized that timely and personalized feedback can significantly improve student understanding and engagement. Mojjada, Chand, and Lakshmi Priyanka (2024) discussed the evolution of AI virtual tutors in higher education and noted that intelligent tutoring platforms are increasingly being designed to act as continuous academic support systems rather than simple question-answer tools.Yakubu, Sain, and Lawal (2025) studied student perceptions of AI as a virtual tutor and found that learners generally respond positively to AI-based instructional assistance when the system is easy to use, supportive, and academically focused. Joy, Santhosh, and Ansari (2024) explained that AI tutoring environments can enhance self-learning by combining content delivery, question answering, and guided explanation within the same platform. Arun, Akash, and Augustin (2025) explored the idea of AI tutors in immersive digital environments, indicating that future tutoring systems may combine conversational intelligence with rich visual and interactive elements. Collectively, these studies confirm that AI tutoring systems are valuable for modern education, but many existing solutions still lack integrated support for multimodal input, session continuity, topic-wise progress tracking, and lightweight deployment. The present work aims to address these gaps through a practical web-based tutoring framework.

III. METHODOLOGY

The methodology of the proposed system follows an application-centric pipeline for delivering real-time educational assistance. The system is organized into sequential functional stages beginning with user interaction, followed by input processing, context construction, AI response generation, and learning-support updates. This flow enables the tutor to operate as a guided instructional assistant rather than a simple response generator.At the initial stage, the learner logs into the platform and submits a doubt through text, voice input, PDF upload, or image upload. The frontend captures the request and forwards it to the backend through an API route. The backend first identifies the active session and loads previous conversation history from the SQLite database. This helps preserve continuity and allows the tutor to respond with contextual awareness. If the user uploads a PDF, the system extracts textual content from the document using the pypdf library. If the user uploads an image, the backend sends the image to a multimodal language model to obtain a description relevant to the tutoring context. The student query, prior session history, and any file-derived content are then combined into a structured prompt. A tutor persona prompt defines the expected teaching style, including simple explanation, step-by-step guidance, analogies, encouragement, and topic tagging. The complete prompt is submitted to a Groq-hosted large language model, which generates a response aligned with the tutoring objective. The result is returned to the frontend, stored in the database, and used to update supporting features such as topic progress, speech output, and quiz preparation. This methodology ensures that the system remains contextual, multimodal, persistent, and educationally oriented throughout the interaction cycle.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 17, April 2026
Impact Factor: 8.2

IV. PROPOSED SYSTEM ARCHITECTURE

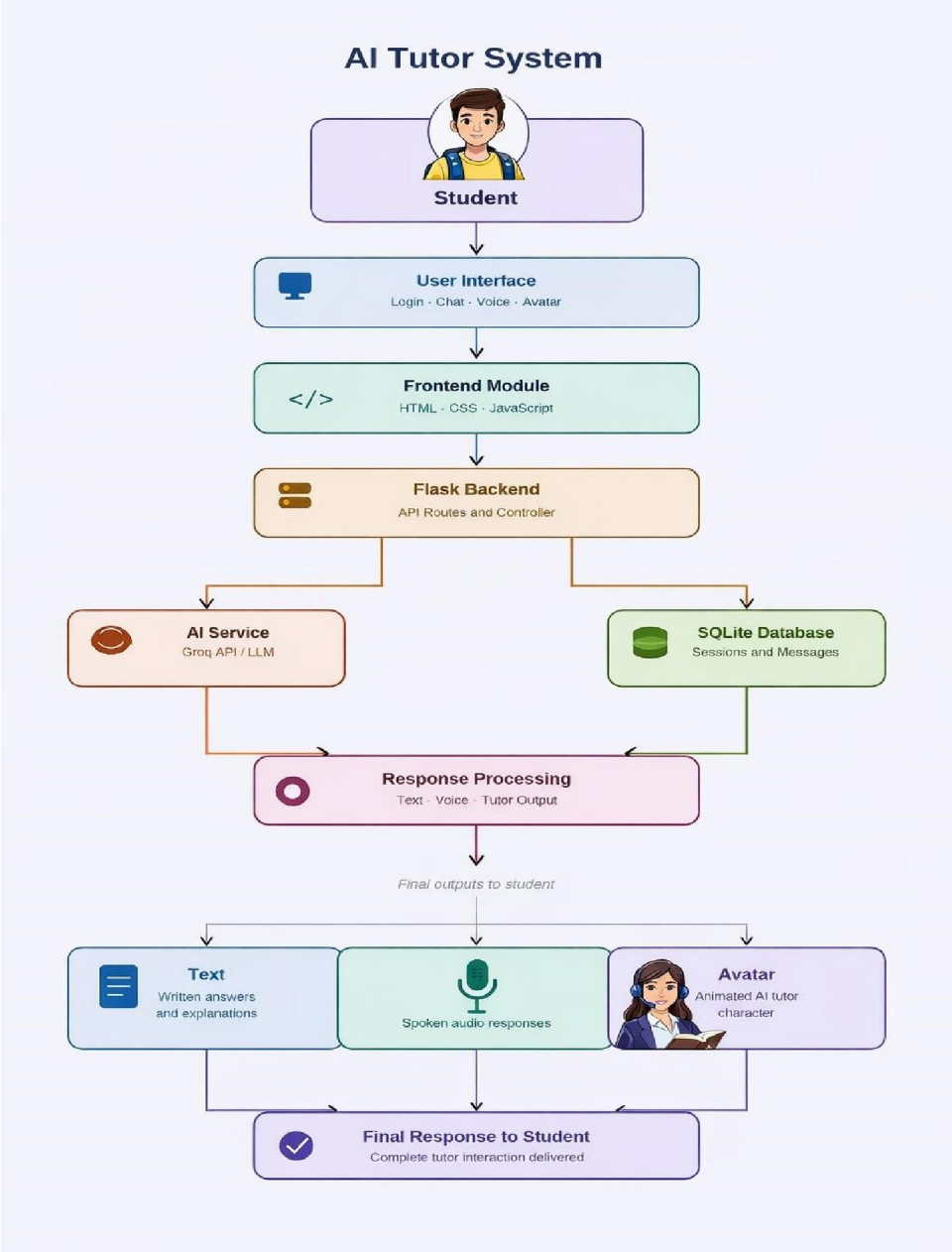

Figure 1: Architecture Daigram

V. MODEL DESCRIPTION

1. User Query Acquisition

The proposed AI Tutor system begins its operation by collecting input from the student through multiple interaction modes, namely text input, voice input, PDF upload, and image upload. This multimodal query acquisition mechanism is one of the major strengths of the system because it makes the platform more flexible, inclusive, and suitable for

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 17, April 2026 Impact Factor: 8.2

different learning preferences. Some students may prefer typing their doubts, while others may feel more comfortable speaking their questions or uploading study materials directly.When a user enters a question in text form, the frontend captures the message and forwards it to the backend for further processing. In the case of voice input, the browser's speech recognition feature converts the spoken query into textual form before it is submitted. If the learner uploads a PDF, the system extracts the relevant text from the document, while image uploads are processed through a multimodal model to derive a descriptive interpretation. By supporting all these input forms, the system ensures that students can interact with the tutor in a natural and convenient manner.This stage is essential because it determines how effectively the system can understand the learner's requirement.

2. Prompt Engineering

Prompt engineering is the central instructional mechanism that shapes the tutor's behavior. Instead of relying on a custom-trained educational model, the proposed system uses a carefully designed prompt to control how the large language model responds to students. In other words, the prompt acts as the pedagogical policy of the tutor.The system prompt is structured to define the role of the AI assistant as a warm, patient, and academically focused virtual tutor. It restricts the domain of discussion to AI-related subjects such as Artificial Intelligence, Machine Learning, Deep Learning, Natural Language Processing, Neural Networks, and Data Science. This prevents irrelevant or open-domain responses and helps maintain subject-specific academic quality. In addition, the prompt enforces a teacher-like response style. The model is instructed to explain concepts step by step, use simple language, include analogies when necessary, and maintain an encouraging tone throughout the conversation.

3. Large Language Model for Text Tutoring

The primary language model used in the proposed system for conversational tutoring is llama-3.3-70b-versatile, accessed through the Groq API. This model forms the core reasoning and explanation engine of the platform. Whenever a student submits a question, the system combines the current query with the stored conversation history and any relevant extracted file content, then sends the full context to the model for response generation.The chosen model is responsible for understanding the learner's doubt, generating a clear explanation, producing examples, responding in a conversational tone, and even supporting revision features such as quiz generation. Since it is a powerful pre-trained foundation model, it provides high-quality reasoning and language generation without the need for large local computation resources.

4. Multimodal Image Understanding

To handle image-based queries, the system uses the multimodal model meta-llama/llama-4-scout-17b-16e-instruct. This model is particularly useful when students upload diagrams, handwritten notes, screenshots, or visual representations related to AI and computing concepts. Since the main tutoring workflow is text-based, the role of the multimodal model is to convert image content into meaningful textual context. When an image is uploaded, it is encoded and sent to the multimodal model along with a domain-focused instruction asking for a description relevant to AI or technical learning. The generated description is then appended to the student's query and passed into the main tutoring context. This makes it possible for the tutor to explain the image content, clarify diagrams, and connect visual material with theoretical concepts.

5. PDF Text Extraction

The PDF support feature is implemented using the pypdf library. This module enables the system to read uploaded
PDF files and extract textual content from them. The extracted text is then merged with the learner's query before being submitted to the language model. This allows the tutor to generate answers not only based on the student's question, but also based on the actual document content. This functionality is particularly important in academic environments where students frequently use digital notes, assignment files, lecture materials, and reference documents in PDF format. Instead of manually copying content from such documents, the learner can upload the file directly and ask for explanations, summaries, or clarifications. The PDF extraction module improves contextual relevance and makes the tutoring system more useful for real-time study support. It extends the system beyond simple question answering and enables document-assisted learning, which is highly valuable for academic doubt resolution.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 17, April 2026 Impact Factor: 8.2

6. Speech Recognition and Speech Synthesis

The proposed system includes both speech recognition and speech synthesis in order to support a more interactive and natural tutoring experience. Speech recognition allows the learner to ask questions through voice input using browserbased APIs such as Speech Recognition or web kit Speech Recognition. The spoken utterance is automatically converted into text and inserted into the query field for submission On the output side, speech synthesis is used to read the tutor's responses aloud using the browser's built-in speech Synthesis interface. This allows students to listen to explanations instead of only reading them on the screen. Such a feature is beneficial for auditory learners and also improves accessibility for users who may find voice-based interaction more convenient. Together, these two technologies make the virtual tutor feel more conversational and human-like. They reduce the friction of interaction and help transform the system from a static chatbot into a more engaging educational assistant.

7. Topic Extraction and Quiz Generation

The system includes a mechanism for topic extraction in order to monitor learning progress and support quiz-based reinforcement. At the end of each response, the tutor includes a topic marker generated by the model as instructed through the prompt. The frontend then extracts this marker using pattern matching and stores it as part of the learner's covered topics. These tracked topics are displayed in the progress panel and later used for quiz generation. When the learner requests a quiz, the system selects one or more previously learned topics and prompts the language model to generate a multiple-choice question. The generated response is parsed and displayed interactively on the frontend so that the student can test their understanding immediately.

8. Session Persistence

The proposed AI Tutor uses SQLite for session persistence and conversation storage. Every session, along with user messages, assistant responses, and timestamps, is stored in the database. This enables the system to reload previous discussions whenever the user returns to the application.Session persistence plays a crucial role in maintaining conversational continuity. Without storage, every question would be treated as an isolated request. By preserving the history of interactions, the system can provide more contextual responses and build upon earlier explanations. This creates a smoother and more personalized learning experience for the student. In addition, stored conversations serve as a practical record of the learner's progress. Students can revisit earlier doubts, review previous explanations, and continue learning from where they left off. Therefore, the SQLite persistence module strengthens both usability and educational continuity.

VI. OUTPUTS AND ANALYSIS

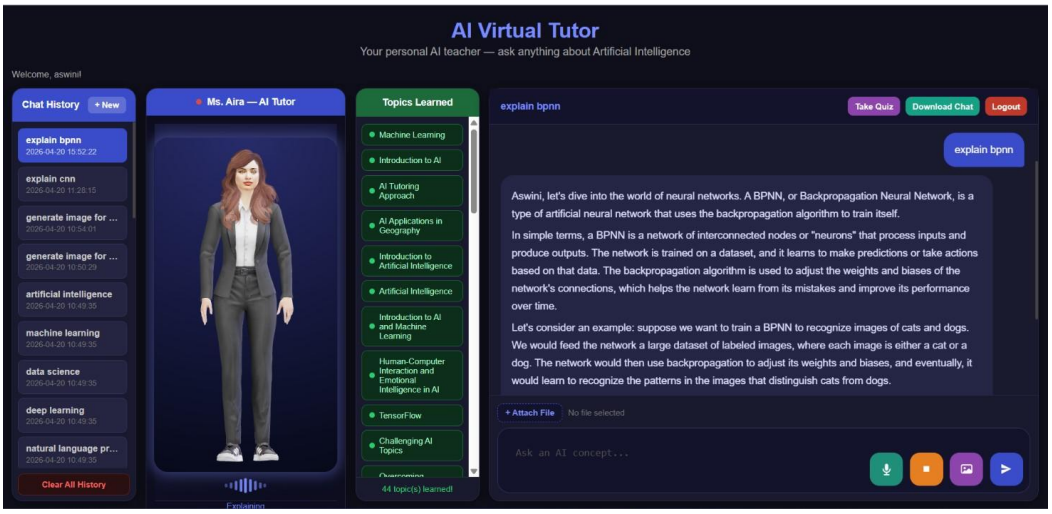

Figure 2 :Text tutoring output

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 17, April 2026 Impact Factor: 8.2

The system successfully supports standard text queries related to ai concepts. The stored sessions show a variety of conceptual prompts indicating that the system produces topic-aligned educational responses

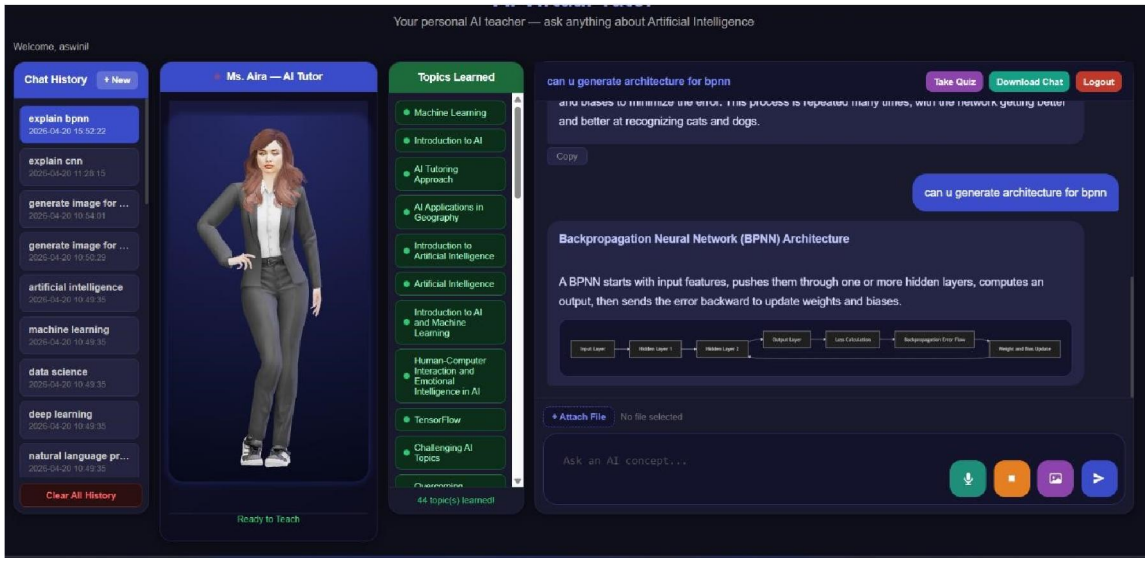

Figure 3: Architecture diagram output

The image generation model successfully produces relevant visual content based on the user's query or concept. Generated images help in improving understanding by providing a visual representation of abstract or complex topics

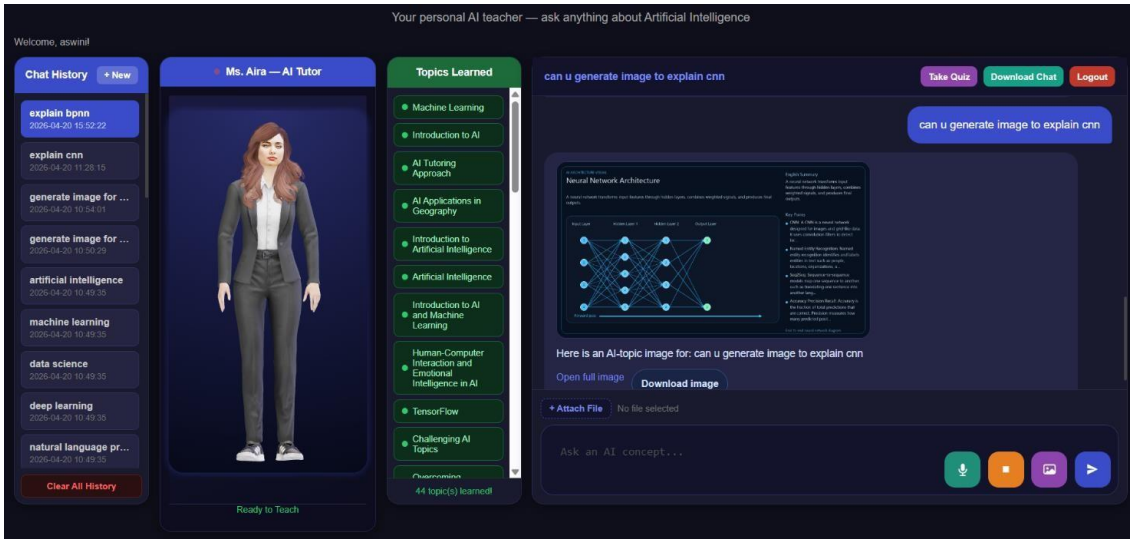

Figure 4: Image generation output

The image generation model successfully produces relevant visual content based on the user's query or concept. Generated images help in improving understanding by providing a visual representation of abstract or complex topics.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 17, April 2026
Impact Factor: 8.2

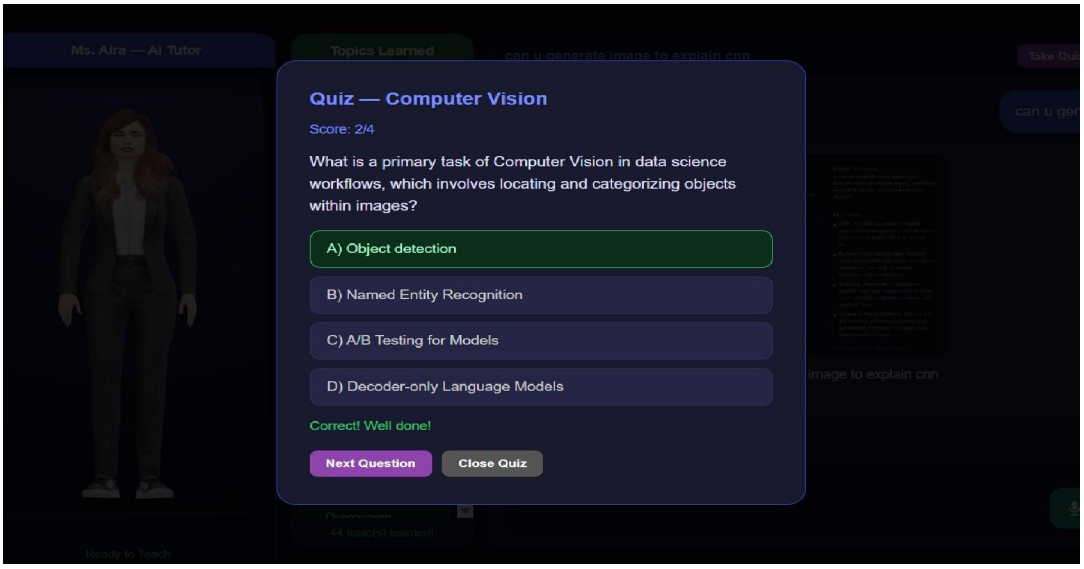

Figure 5:Quiz generation output

The quiz module dynamically selects one learned topic and asks the backend to generate a multiple-choice question. The frontend parses the question format and allows the learner to answer interactively. The score is maintained in memory.

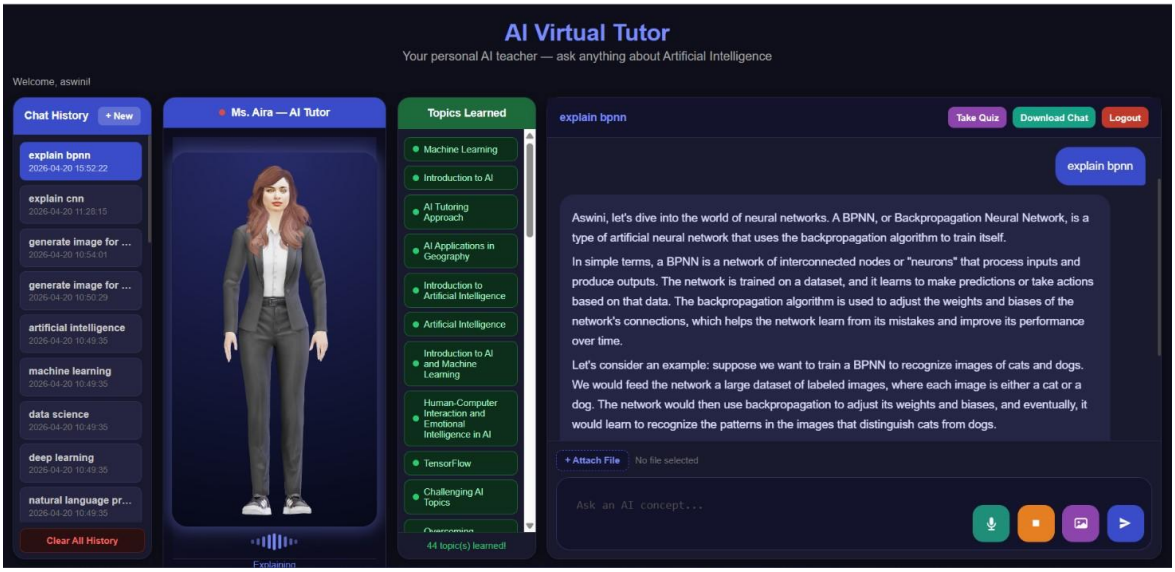

Figure 6**:** File Input result

The system supports two main file types:

PDF

Image

For PDFs, the backend extracts text and forwards it to the main LLM. For images, a multimodal model is used to generate a description relevant to AI or technology concepts. This allows the tutor to respond with context drawn from the uploaded content.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 17, April 2026
Impact Factor: 8.2

VII. CONCLUSION

This work presented an AI-powered virtual tutor for real-time student doubt resolution in Artificial Intelligence and related subjects. The system successfully integrates a web-based frontend, Flask backend, Groq-hosted large language models, multimodal image support, PDF text extraction, speech interaction, topic tracking, quiz generation, and SQLite-based session memory into a unified educational platform.The proposed tutor addresses the limitations of static learning resources by offering immediate, contextual, and teacher-like explanations. Through prompt engineering and persistent interaction history, the system delivers structured and supportive responses that improve the learning experience. Its multimodal design further increases usability by allowing students to interact through text, speech, images, and documents.In conclusion, the project demonstrates that a practical and effective intelligent tutoring system can be built using lightweight web technologies and pre-trained foundation models without the need for costly local training infrastructure. The developed system can serve as a strong foundation for future educational applications that aim to provide personalized, accessible, and interactive academic support.

REFERENCES

[1]. Pushpa, D., Arpitha, H. M., and Ananya, M., "AI-Powered Virtual Tutor with Generative AI," International Research Journal of Modernization in Engineering, Technology and Science (IRJMETS), vol. 7, no. 12, 2025.
[2]. Kestin, G., Miller, K., Klales, A., Milbourne, T., and Ponti, G., "AI Tutoring Outperforms In-Class Active Learning: A Randomized Controlled Trial," Scientific Reports, 2025.
[3]. Mojjada, H., Chand, N. G., and Lakshmi Priyanka, "The Evolution of AI Virtual Tutors in Modern Higher Education," International Journal of Innovative Research in Technology (IJIRT), vol. 11, no. 7, 2024.
[4]. Yakubu, M. A., Sain, Z. H., and Lawal, "Student Perceptions of AI as a Virtual Tutor," Indonesian Journal of Progressive Pedagogy, vol. 1, no. 1, 2025.
[5]. Joy, E. M., Santhosh, A., and Ansari, "AI as a Tutor," International Research Journal of Engineering and Technology (IRJET), vol. 11, no. 4, 2024.
[6]. Arun, R., Akash, S. V., and Augustin, "AI Tutor in a 3D Environment like Metaverse," International Journal of Creative Research Thoughts (IJCRT), vol. 13, no. 5, 2025.
[7]. "Individualized AI Tutor Based on Developmental Learning Networks," Google Scholar indexed reference source.