

Edu Tutor AI: Personalized Learning

Generative AI with IBM

1.Introduction

- Title : AI Study Helper
- Team member : Jananee R
- Team member :Saroja M
- Team member :Deepshika M A
- Team member :Rithika V

2.Project overview

Purpose :

The purpose of this project is to **enhance the learning experience using Artificial Intelligence**. Traditional learning often requires constant guidance from teachers and access to multiple resources. With the rise of Generative AI, students can now receive instant explanations, practice quizzes, and interactive learning support.

This project aims to:

- Provide **on-demand explanations** of academic concepts in simple and detailed language.
- Help students **self-assess** their understanding through automatically generated quizzes.
- Reduce the workload of educators by offering an **AI-powered tool** for content and quiz creation.
- Create a **user-friendly educational assistant** that can be accessed by anyone, anywhere, using Google Colab or a local machine.

In short, the project's purpose is to **make learning smarter, faster, and more engaging** by combining the power of **Generative AI with an interactive interface**.

Features:

- **Concept Explanation** → Enter any topic and get a detailed explanation with examples.

- **Quiz Generator** → Automatically generates 5 quiz questions (MCQ, True/False, Short Answer) with an answer key.
- **User-Friendly UI** → Built using Gradio for easy interaction.
- **Runs Anywhere** → Works in Google Colab, Jupyter Notebook, or locally.

3. Architecture

The **Educational AI Assistant** follows a simple yet effective architecture that integrates a **user interface**, a **language model**, and **response generation logic**.

Workflow Steps

User Input

- User enters a concept (for explanation) or a topic (for quiz generation) in the Gradio interface.

Tokenizer

- The input is converted into token IDs using HuggingFace's AutoTokenizer.

Model Processing

- The encoded input is passed to **IBM Granite 3.2-2B Instruct** model.
- The model generates a response based on the prompt (either detailed explanation or quiz).

Response Handling

- The output is decoded back into human-readable text.
- Any extra tokens/special characters are cleaned.

Display to User

- The response is shown in Gradio's output box under the correct tab.

4. Setup Instructions

To use the Educational AI Assistant, follow these steps:

Download or Clone the Project

- Get the project files from GitHub by downloading them as a ZIP file or cloning the repository.

Install Required Software

- Make sure you have Python installed on your system.
- Install the necessary libraries: Gradio, Torch, and Transformers.

Set Up the Environment

- (Optional) Create a virtual environment to keep the project dependencies separate.
- Activate the environment before running the project.

Run in Google Colab (Recommended for Beginners)

- Open the project notebook in Google Colab.
- Change the runtime to GPU for faster execution.
- Run all the cells to launch the application.

Run Locally (Optional)

- Open the project in your code editor or terminal.
- Run the application file.
- A link will be generated by Gradio, which you can open in your browser to access the assistant.

GPU Support

- For faster response generation, enable GPU in Google Colab or use a system with CUDA-enabled GPU if running locally.

5. Folder Structure

- **app.ipynb** → Main notebook for running the project in Google Colab.
- **app.py** → Standalone script to run the project locally.
- **requirements.txt** → Dependencies required for installation.
- **README.md** → Documentation and instructions for GitHub repository.
- **docs/** → Stores detailed project documentation or reports.
- **outputs/** → Contains sample outputs generated by the assistant.
- **assets/** → Stores images, diagrams, or screenshots used in the project report.
-

6. Running the Application

- i) Open the project folder on your computer.
- ii) Start the application file using your Python environment.

- iii) Once the application starts, Gradio will provide a **local web link**.
- iv) Copy the link into your browser to access the assistant.
- v) If sharing is enabled, a **public link** will also be generated for others to use.

7. Authentication

At present, the **Educational AI Assistant** does not require user login. However, authentication can be integrated to ensure that only authorized users (e.g., teachers, registered students) can access the application.

◆ Possible Authentication Methods

1. Gradio Built-in Authentication

- Gradio provides a simple way to protect the interface with a username and password.
- Example: When launching the app, add login credentials so only authenticated users can open the interface.

2. Token-Based Authentication

- Assign unique tokens (or keys) to each user.
- Users must enter a valid token before accessing the assistant.

3. OAuth / Google Authentication (Advanced)

- Integrate with Google, Microsoft, or institutional login systems.
- Students and teachers can log in with their official accounts.

◆ Benefits of Authentication

- Prevents unauthorized access to the learning assistant.
- Keeps usage limited to a particular group (classroom, institution, or project team).
- Enables tracking of student activity and quiz attempts (if extended with user profiles).

8. User Interface

The Educational AI Assistant provides a simple and user-friendly interface built using Gradio. The interface is web-based, meaning users can access it directly through a browser without installing additional software.

◆ Layout Overview

- The application is divided into two main tabs:

1. Concept Explanation Tab

- Input box: Allows the user to type any concept (e.g., "Machine Learning").
- Action button: "Explain" to generate the explanation.
- Output area: Displays a detailed explanation with examples.

2. Quiz Generator Tab

- Input box: Allows the user to type a topic (e.g., "Physics").
- Action button: "Generate Quiz" to create quiz questions.
- Output area: Displays five questions in different formats (MCQ, True/False, Short Answer) along with answers.


9. Testing

Testing was carried out to ensure that the **Educational AI Assistant** functions correctly, produces accurate outputs, and provides a smooth user experience. Both **functional testing** and **usability testing** were performed.


◆ 1. Functional Testing

This verifies that all features work as intended.

• Concept Explanation


- Input: "Machine Learning"
- Expected Output: A detailed explanation with examples.
- Result:  Correct explanation generated.

• Quiz Generator

- Input: "Physics"
- Expected Output: Five quiz questions in mixed formats with answers.
- Result:  Quiz generated successfully.

• Gradio Interface

- Tabs load correctly.

- Buttons trigger the correct functions.
- Input and output boxes work without errors.
- Result:  Passed.

◆ 2. Usability Testing

Testing with users (students and teachers) to check ease of use.

- The interface is **easy to navigate** with clear labels.
- Explanations are **easy to understand** and include examples.
- Quizzes provide **variety** (MCQ, True/False, Short Answer).
- Users reported that the tool is **helpful for revision and practice**.

◆ 3. Performance Testing

- Tested in Google Colab with and without GPU.
- GPU significantly improves response time.
- On CPU, response generation is slower but still functional.

◆ 4. Limitations Found During Testing

- Responses may sometimes include repetitive or lengthy text.
- Quiz quality depends on the model's knowledge and prompt design.
- Requires internet connection to load the model.

10. Future enhancement

For educational institutions, the assistant can be integrated with a Learning Management System (LMS) such as Moodle or Google Classroom, where authentication is handled automatically through student accounts.

11. Conclusion

The Educational AI Assistant successfully demonstrates how Generative AI can be applied in the field of education to make learning more interactive and engaging. By combining the IBM Granite 3.2-2B Instruct model with a Gradio-based user interface, the system provides two major functionalities:

1. Concept Explanation – Students can instantly receive clear, detailed, and example-based explanations of academic topics.
2. Quiz Generation – The system automatically generates quizzes with varied question formats, helping learners test and improve their understanding.

During testing, the assistant proved to be user-friendly, reliable, and effective for both self-learning and teaching support. While there are limitations such as dependency on internet connectivity and occasional variation in responses, the project has shown strong potential as a digital learning companion.

This project highlights the possibilities of AI-powered education, where learning is no longer restricted to textbooks and classrooms but is accessible anytime, anywhere. With future enhancements like voice input, multilingual support, and integration with Learning Management Systems (LMS), the Educational AI Assistant can evolve into a powerful tool for students, educators, and lifelong learners.