TECHNICAL REPORT

AI CLASSROOM ASSISTANT

TEAM NAME : VIVID SMART DRONA

DATE OF SUBMISSION : 12/07/2025

Abstract

The AI Classroom Assistant is a cutting-edge multi-modal teaching tool that uses artificial intelligence to integrate speech, text, and image understanding, revolutionising the classroom experience. This system, which was created with Python and a carefully selected library stack, allows educators and learners to engage with instructional materials in a natural way. Using a local vision-language model, the assistant creates concise, understandable explanations based on spoken queries, typed prompts, or photos of diagrams. This project integrates language modelling, text-to-speech, speech-to-text, and optical character recognition (OCR) into a single, integrated platform. In order to improve comprehension and close learning gaps, the main goals are to make difficult subjects approachable and create an engaging learning environment that stimulates interest and involvement.

Introduction

Modern classrooms, both virtual and real, have many challenges to ensure that each student understands a complicated subject. Most students find it difficult to cope when explanations are text and audio-based only, especially for subjects that often come with technical charts or complex vocabulary. The instructor may repeatedly give explanations to fit various speeds of learning, which hinders the general pace of the class.

This AI Classroom Assistant confronts these challenges head-on by allowing students and instructors to engage with learning material in different fashions. A query may be phrased by talking into a mic, typed into a basic input field, or even grabbed from a slide or written note via a snipping tool. The system responds in clean, simplified text, either as on-screen text or as synthesized speech read aloud. The assistant is developed in a lightweight, cross-platform environment based on Python's Tkinter for the graphical interface to run on common classroom machines. By supporting various input and output modes, it assists in reaching visual, auditory, and kinesthetic learners, promoting an egalitarian atmosphere where none of the learners are left behind.

Problem Statement

Modern classrooms often lack intelligent, real-time support tools that cater to individual student needs without interrupting the teaching flow. Students may hesitate to ask questions or struggle to understand complex visuals during lectures. To address this, the proposed solution is a multimodal AI-powered classroom assistant that enables students to interact using text, voice, or visual inputs. By integrating OCR, speech recognition, and large language models like LLaVA, the assistant can provide instant, personalized explanations for diagrams, slides, or spoken questions. This enhances student engagement, fosters independent learning, and bridges the gap between instructional delivery and real-time understanding.

Motivation Behind the Project

Education statistics regularly point to gaps in class participation and understanding. A lot of students are reluctant to pose questions due to shyness or fear of being seen as lacking information. In such instances, flowcharts on slides or whiteboards tend to go unchecked in detail, with the result that important gaps in understanding go unnoticed. Further, with the widespread expansion of online education, teachers must serve many students with varying learning styles and rates without always possessing suitable tools.

This project grew out of the necessity to lower such barriers. By giving the learner an assistant that makes it possible to ask questions by way of natural speech, typed queries, or actual captures of diagrams, we hope to help make learning more accessible. By incorporating an AI language model that adjusts its explanations in order to be concise and simple, we give learners greater control to learn according to their own abilities. By doing this, the project not only encourages students to overcome their fears but also helps teachers by taking away repeated explanations, allowing them to concentrate more on more complex teaching issues. Overall, this makes a more interactive and efficient classroom.

Data Source

Unlike typical data-based machine learning initiatives relying on massive, stagnant datasets, this helper works with dynamic, real-time data input from users as it learns. The voice module records sound from a microphone and transcribes it into text through Google's Speech Recognition service, which parses real-time spoken queries. The snipping tool enables users to clip any area of the screen, e.g., a section of a slide or a hand-written note, and save it as an image file. The image is processed through the Tesseract OCR to recognize any textual information. Apart from textual and speech inputs, these images are also passed on to a multi-modal language model so that the AI can comprehend and explain diagrams in context.

All this information is temporary and attached to the unique learning session, providing a customized experience. The language comprehension itself is powered by the locally installed LLaVA model accessed via the Ollama platform, which is provided the text prompt and, where supplied, encoded image information. The assistant uses these in-real-time inputs instead of pre-gathered data so it is extremely flexible to any topic, language, or classroom setting.

Work

The project incorporates multiple technical modules into one system, all of which play an essential role in providing an interactive learning experience. The speech recognition part at its heart employs the speech_recognition library to record voice input. It actively adapts to background noise and listens for a certain amount of time, transcribing spoken questions into text and directly feeding it into the AI query system. The optical character recognition module uses Tesseract, accessed via the pytesseract Python wrapper, to read text from images taken by the in-house snipping tool. This in-house snipping tool itself is a Tkinter overlay that allows users to drag and select any part of the screen, which gets automatically saved and processed.

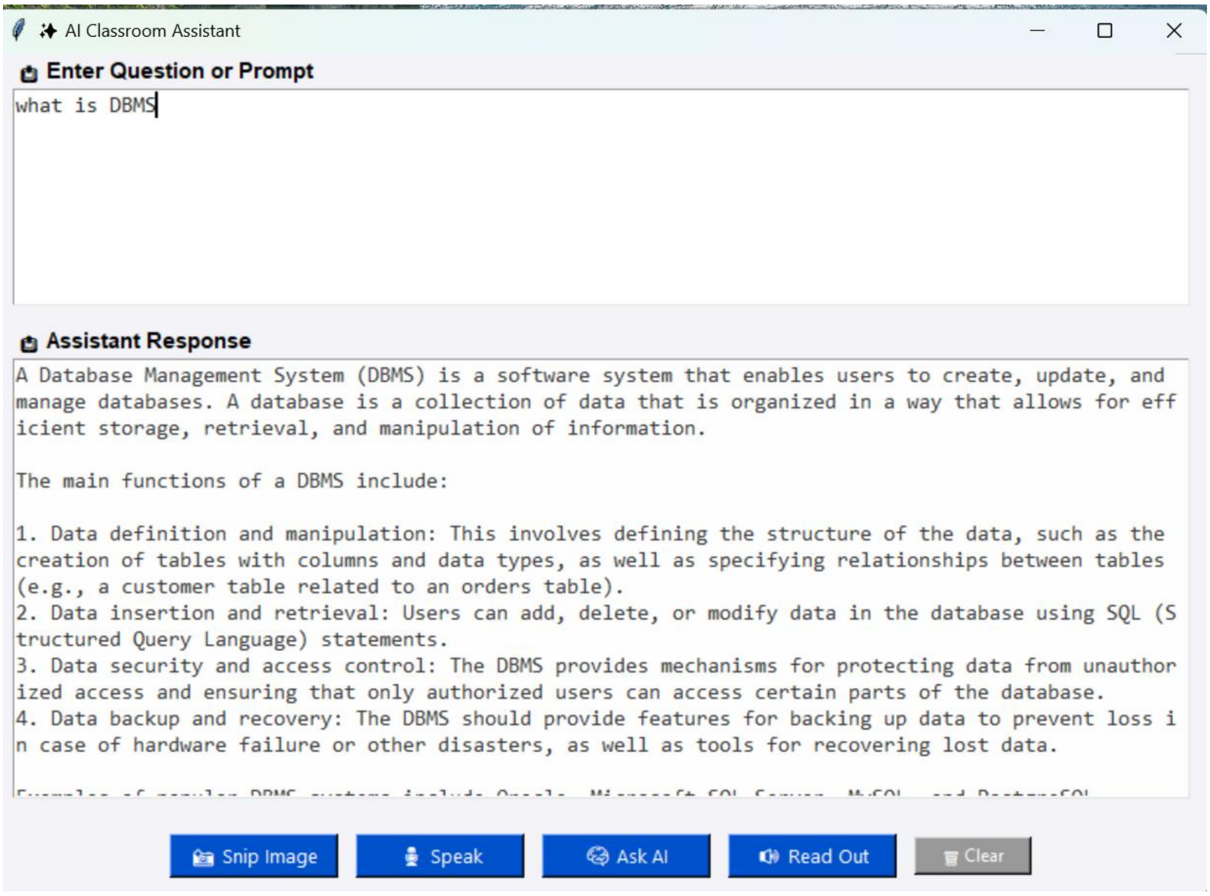
The LLaVA model offers natural language comprehension and explainability, which is hosted locally through Ollama. This is done to eliminate cloud service dependence and ensure privacy, while speed and dependability are also enhanced. The assistant provides good prompts to this language model, with or without base64-encoded images, to obtain context-specific explanations. To assist auditory learners and to provide accessibility, the project also has a text-to-speech module implemented using pyttsx3, which speaks the responses of the AI.

All these functionalities are combined in an easy-to-use interface built using Tkinter. The GUI has input and output text boxes and a collection of buttons to manage speaking, snipping, querying the AI, reading the output out loud, and clearing the chat. Threads are employed judiciously so that time-consuming operations, such as speech recognition or AI queries, do not freeze the interface, providing a responsive and smooth user experience.

Contribution

Name	Contribution
1. Singireddy Sriharsha	- Multimodal Input Processing (Text, Voice, and Image)
2. Nune Vyshali	- Real-Time AI Querying Using LLaVA via Ollama
3.Sunkoju Deekshith Chary	- Interactive GUI Built with Tkinter

Result



```
what is DBMS
```

```
A Database Management System (DBMS) is a software system that enables users to create, update, and
manage databases. A database is a collection of data that is organized in a way that allows for eff
icient storage, retrieval, and manipulation of information.

The main functions of a DBMS include:

1. Data definition and manipulation: This involves defining the structure of the data, such as the
creation of tables with columns and data types, as well as specifying relationships between tables
(e.g., a customer table related to an orders table).
2. Data insertion and retrieval: Users can add, delete, or modify data in the database using SQL (S
tructured Query Language) statements.
3. Data security and access control: The DBMS provides mechanisms for protecting data from unauthor
ized access and ensuring that only authorized users can access certain parts of the database.
4. Data backup and recovery: The DBMS should provide features for backing up data to prevent loss i
n case of hardware failure or other disasters, as well as tools for recovering lost data.

Examples of popular DBMS systems include Oracle, Microsoft SQL Server, MySQL, and PostgreSQL
```

Result Link: https://drive.google.com/file/d/1M1hJch8ybpqH6pAb__Az60nN7NFb6yui/view?usp=drive_link

Github link : https://github.com/sriharsha2005reddy/Intel-Unnati-