MAHATMA GANDHI INSTITUTE OF TECHNOLOGY

GANDIPET, HYDERABAD - 500075

PROJECT DOCUMENTS SUBMITTED FOR INTEL UNNATI INDUSTRIAL TRAINING

MAY-2025 TO JULY-2025

DEPARTMENT OF EMERGING TECHNOLOGIES

Intel Unnati Industrial Training Program 2025

INTEL UNNATI SUMMER TRAINING–2025

Mentor: Dr.T.V. RAJINIKANTH, Professor & Convener R&D

DEPARTMENT OF ET

MAHATMA GANDHI INSTITUTE OF TECHNOLOGY , GANDIPET, HYDERABAD

E-Mail: rajinikanthtv_cse@gmail.com

Participated Student list and selected for Certificate Awardees

S. No	Problem Statement	Team Name	Team Size	Team Lead & Team Members	Mentor Name
2	AI-Powered Interactive Learning Assistant for Classrooms	Yukti Mind	4	RADHI T V J	Dr.T V RAJINIKANTH
				KONGARI BHUVANA	
				AMAN ADUSUMILLI	
				AASHRITH RAJ	

PROJECT TITLE: AI-Powered Interactive Learning Assistant for Classrooms

TEAM NAME: Yukti Mind

TEAM MENTOR: Dr.T. V. Rajinikanth

Professor & Convener R&D,

Dept. of ET

rajinikanthtv_cse@mgit.ac.in

INTEL MENTOR: Debdyut Hajra,

debdyuthajra@rediffmail.com

TEAM MEMBERS:

1. Radhi T V J Rollno:22261A6648,CSE-AIML(Team Lead) (ET)
2. Kongari Bhuvana Rollno:22261A6627,CSE-AIML(ET)
3. Aman Adusumilli Rollno:22261A6601,CSE-AIML(ET)
4. Aashrith Raj Rollno: 22BCS047 ,CSE

COLLABORATIVE PROJECT WITH INTEL

PROJECT TITLE : "AI-Powered Interactive Learning Assistant for Classrooms"

Team Name : Yukti Mind

Team Mentor : Debdyut Hajra

Internal Mentor : Dr T V RAJINI KANTH, Professor & Convener

tvrajinikanth_cse@mgit.ac.in

Ph.No: 9849414375

Team Members : Radhi TVJ radhitadepalli13@gmail.com CSE-AIML(Team Lead)(ET)

Kongari Bhuvana bhuvanakongari@gmail.com CSE-AIML(ET)

Aman Adusumilli amanadusumilli@outlook.comCSE-AIML(ET)

Aashrith Raj aasrithraj2004@gmail.com CSE

Department Name : Emerging Technologies

Institute Name : Mahatma Gandhi Institute of Technology Gandipet,

Hyderabad- 500075

Date of submission : 12-07-2025

TABLE OF CONTENTS

1. Problem Statement

In traditional classroom and remote learning environments, students and educators face numerous challenges—voluminous content, time-consuming preparation, limited interactivity, and lack of personalized tools for understanding, revision, and assessment. Textbooks, handwritten notes, and scattered study materials are often difficult to synthesize quickly. Educators are burdened with repetitive tasks like test creation and grading, while students struggle to find effective ways to revise and engage with material, especially when under time constraints.

The problem is further compounded in multilingual or hybrid classrooms where diverse learning needs demand adaptive tools. There is a critical need for a comprehensive educational assistant that supports learning, assessment, and engagement—leveraging modern AI technologies to deliver smart, responsive, and multimodal interactions.

2. Solution Approach

The AI-Powered Interactive Learning Assistant provides a centralized, intelligent dashboard where students and teachers can interact with learning materials through text, speech, and visual inputs. The assistant offers:

- Summarization of long documents for quick revision
- Text- and speech-based Q&A over uploaded content
- Automated generation and evaluation of quizzes
- Structured question bank creation
- Interactive flashcards for concept reinforcement
- Diagram analysis using OCR and contextual interpretation

Built using Python and Streamlit, the assistant employs Hugging Face NLP models for language tasks, Tesseract OCR for image-based diagrams, and Google Speech Recognition for voice input. The models are optimized using Intel OpenVINO Toolkit to ensure fast and efficient responses.

3. Novelty of the approach

Unlike existing educational tools that are either static or limited to specific features, this assistant offers a multimodal, all-in-one AI-powered learning platform. Its key innovations include:

- Integration of **text, speech, and visual processing** into a single seamless pipeline
- Document-aware **contextual Q&A** system powered by transformer models
- **Dynamic quiz and flashcard generation** using semantic and syntactic analysis
- **Diagram analysis with OCR + NLP fusion**, enabling users to extract meaning from images
- Deployment-friendly, lightweight design using **Streamlit** for fast interaction
- **Intel OpenVINO optimization** for real-time performance, even on low-end machines

This combination of accessibility, modularity, and responsiveness distinguishes the system from most commercial ed-tech platforms and university tools.

4. Methodology

4.1 System Overview

The application follows a modular architecture designed for scalability and maintainability. The frontend is built using Streamlit, providing an intuitive web-based interface, while the backend leverages Python's robust ecosystem for AI and machine learning operations.

4.2 Technology Stack

Frontend Technologies

- Streamlit: Web application framework for rapid prototyping and deployment
- HTML/CSS: Custom styling and responsive design elements
- JavaScript: Interactive components and dynamic content loading

Backend Technologies

- Python 3.x: Core programming language
- Streamlit: Backend framework integration
- Hugging Face Transformers: NLP model integration
- Intel OpenVINO: Model optimization and acceleration

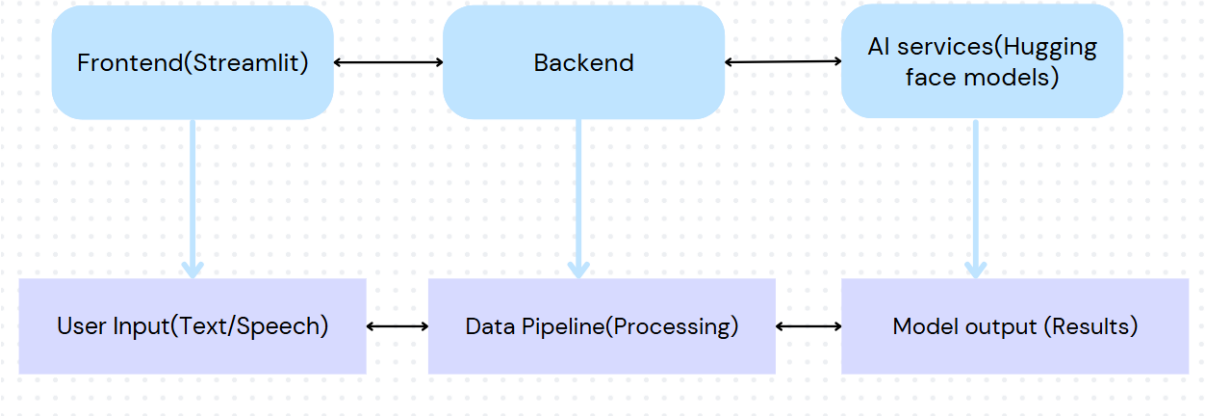
AI/ML Components

- NLP Models: T5-small, DistilBART for text processing
- Speech Recognition: Google Speech Recognition API
- OCR Engine:Tesseract OCR (pytesseract)
- Image Processing: PIL (Pillow)

Data Processing

- PDF Processing: pdfplumber
- Document Processing: python-docx
- Text Processing: NLTK, spaCy

4.3 System Architecture Diagram



4.4 Functional Modules

1. AI Summarization

Utilizes pre-trained transformer models such as T5-small and DistilBART to generate concise, coherent summaries from large chunks of educational text. This module assists in quick content review and comprehension.

2. Q&A Assistant

Employs a contextual Q&A pipeline powered by Hugging Face transformers. It allows users to type or

speak their questions using Google Speech Recognition. The system retrieves relevant answers grounded strictly in the uploaded content, ensuring contextual relevance.

3. Test Generator and Evaluator

Automatically generates 5 MCQs and 5 Fill-in-the-Blank questions using NLP-based keyword extraction and masking. The module scores user inputs and provides instant evaluation, enabling self-assessment.

4. Question Bank Generator

Builds a categorized bank of 1-mark, 2-mark, 3-mark, and 5-mark questions from the input content. This module is optimized for exam preparation and revision planning.

5. Flashcard Generator

Highlights key concepts and generates flashcards containing definitions and explanations, presented in a collapsible format ideal for active recall and spaced repetition.

6. Diagram/Image Analyzer

Uses Tesseract OCR to interpret images containing diagrams or charts. The text extracted from these visuals is contextually matched with document content to generate informative explanations.

5. Describe Advantages and Limitations of the approach

5.1 Advantages:

- **Multimodal Interaction (Text, Speech, Image)**:The assistant allows users to engage through text input, speech commands, and diagram uploads—making it accessible to students with different learning styles and preferences.
- **Contextual Accuracy**:All responses are derived from the uploaded document, ensuring relevance and contextual correctness. This avoids generic AI responses and keeps the content focused on the syllabus or material at hand.
- **End-to-End Automation**:From summarization to question generation and evaluation, the system handles all learning stages automatically—saving time for students and reducing manual effort

for teachers.

- **Interactive and Intuitive UI**:Built using Streamlit, the dashboard is clean, minimalistic, and user-friendly. No prior technical knowledge is required to operate the tool, making it suitable even for school-level users.

- **Performance Optimizations with OpenVINO**:Intel OpenVINO integration enables faster NLP model inference, making even large document tasks highly responsive. This makes the assistant usable on standard laptops and non-GPU machines.

- **Custom Test and Evaluation System**:Students can not only take auto-generated quizzes but also receive instant feedback and scores. Teachers can reuse the question bank generator for test creation.

- **Flashcard-Based Active Recall**:The assistant helps promote memory retention by generating concept-based flashcards in collapsible formats, supporting spaced repetition learning.

- **Scalable and Modular Architecture**:Each module (summarization, Q&A, flashcards, etc.) works independently. This modular design makes the system easy to upgrade, debug, and expand with new features.

- **Deployment Friendly**:Can be deployed both locally and on the cloud, requiring minimal infrastructure—making it ideal for schools, coaching centers, or self-learning platforms.

5.2 Limitations

- **Limited to English (Current Version)**:The system only supports English for NLP tasks. Students using regional languages or multilingual content will need translations or additional modules.

- **OCR Sensitivity to Image Quality**:The diagram analysis feature depends on the quality of uploaded images. Handwritten or low-resolution diagrams may not be accurately processed.

- **Dependence on Cloud APIs**:The speech-to-text functionality currently relies on Google's online API, which requires internet access. Offline or low-bandwidth environments may experience limitations.

- **No Deep Pedagogical Reasoning**: While the assistant generates questions and answers

contextually, it doesn't assess higher-order thinking or cognitive learning levels unless explicitly trained with such models.

- **No Personalization (Yet)**:The current version does not track user history or tailor outputs based on student progress or difficulty level. It treats every input statelessly.

- **Manual Validation Still Required for High-Stakes Use**:While the generated outputs are mostly accurate, test questions and answers must be manually reviewed before formal use in exams or assessments.

- **Not Integrated with LMS Platforms**:Although it can be integrated in the future, there's no direct connection with learning management systems like Moodle or Google Classroom in the current build.

- **Limited Diagram Understanding**:The diagram analysis provides textual explanations based on OCR and document context, but it does not yet support deep visual semantic analysis (like recognizing arrows, shapes, or relationships between objects).

6. Results

The effectiveness of the **AI-Powered Interactive Learning Assistant** is demonstrated through both qualitative user experience and technical performance evaluations. The interactive Streamlit application showcases each module's real-time functionality, with visual outputs and dynamic feedback mechanisms. These confirm the system's utility in enhancing learning, simplifying assessment preparation, and fostering active engagement.

6.1. Output Screens (Qualitative Analysis)

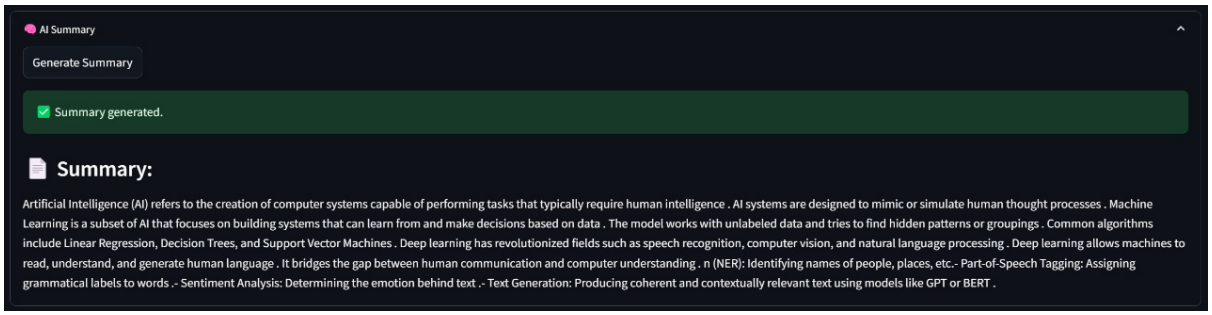
The assistant provides a seamless user interface that allows for visual interaction with every AI module. Below is an overview of the assistant's main features as seen through the live interface:

1. Document Upload and Processing

Users can upload PDF, DOCX, or TXT files. Upon upload, the system extracts and prepares the text.

2. Summarization Output

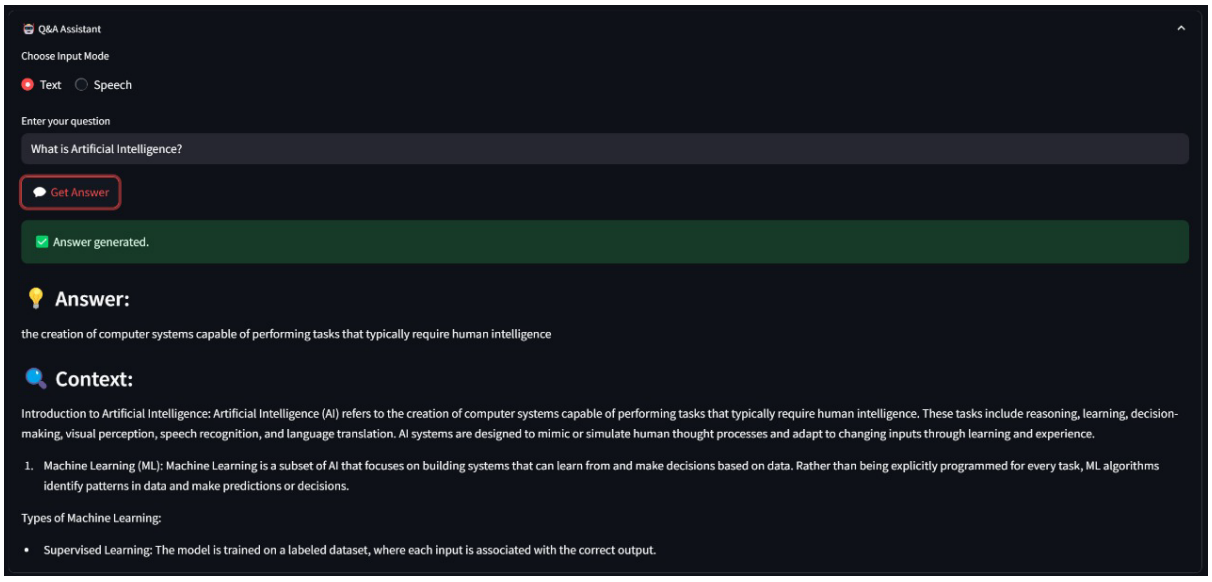
Uploaded content is summarized into key points using Hugging Face models like T5-small.



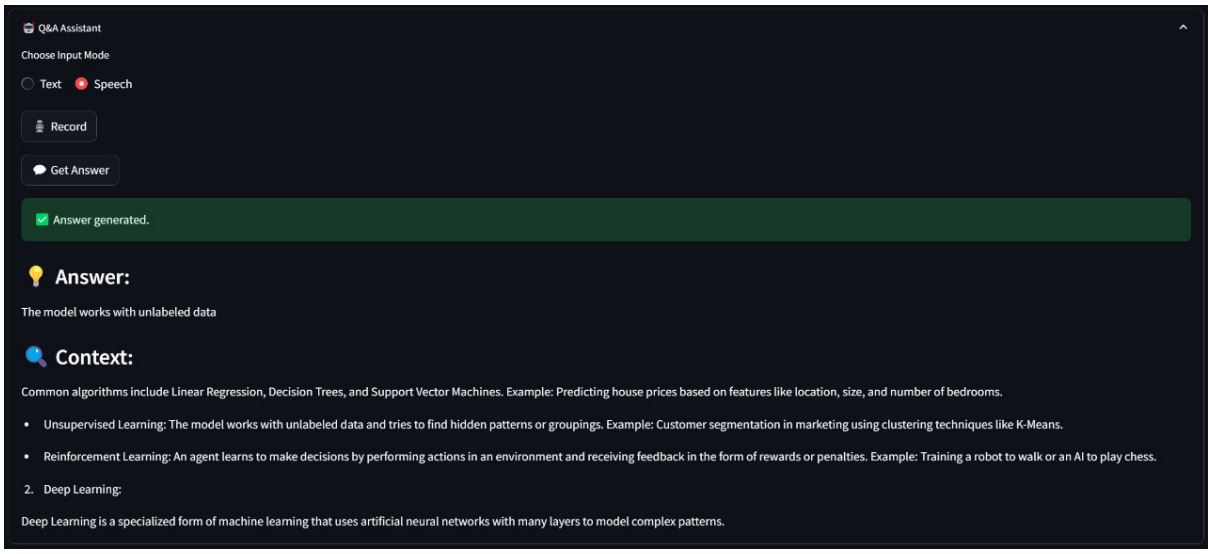
3. Question Answering (Text and Speech)

Users can type or speak their questions using the integrated mic. Responses are generated in context of the uploaded material.

Question answering when text input is given:

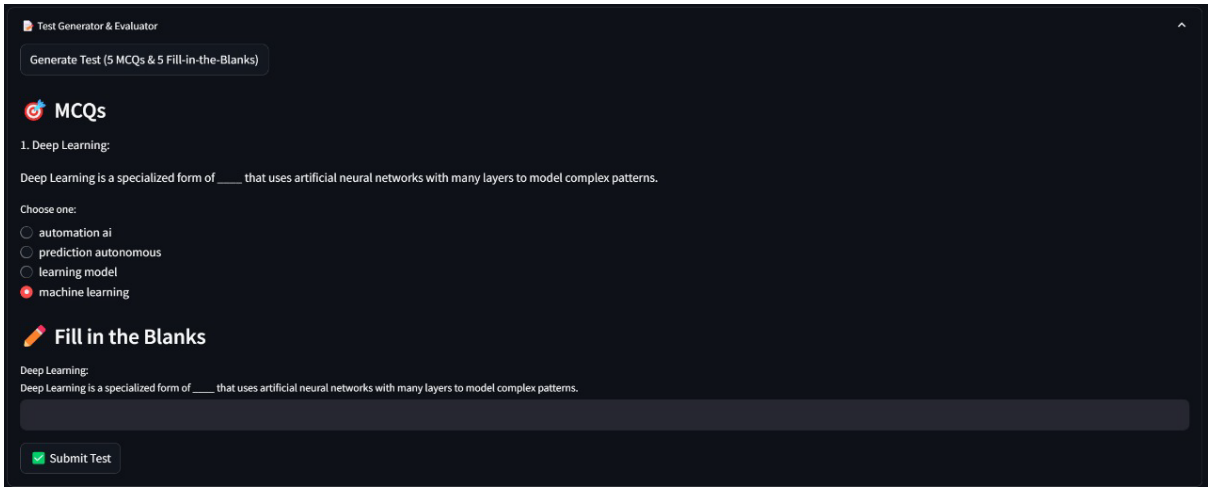


Question answering when speech input is given:



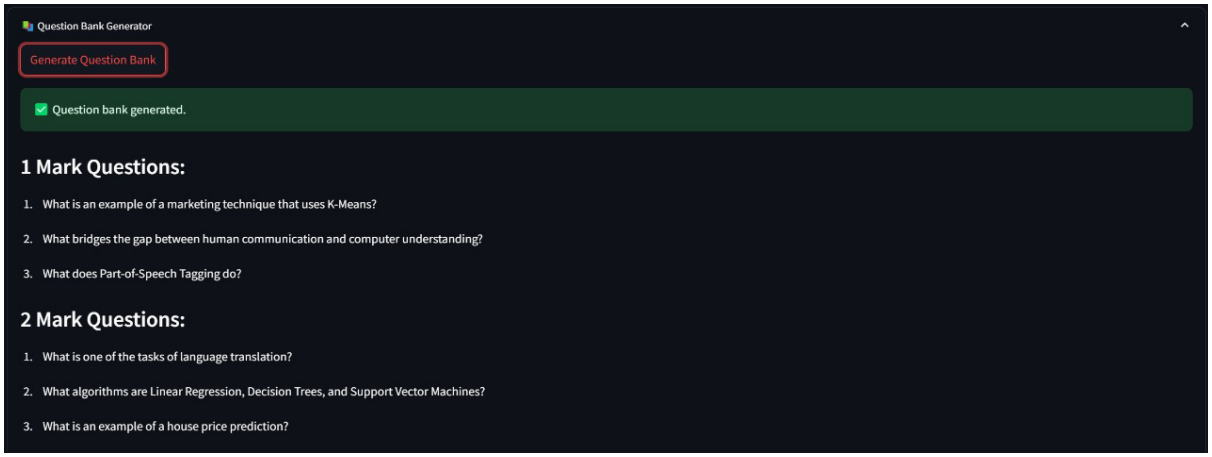
4. Quiz and Test Generation

The assistant generates a set of 10 questions (5 MCQs, 5 Fill-in-the-Blanks) from the content. Users answer directly, and the assistant scores responses in real time.



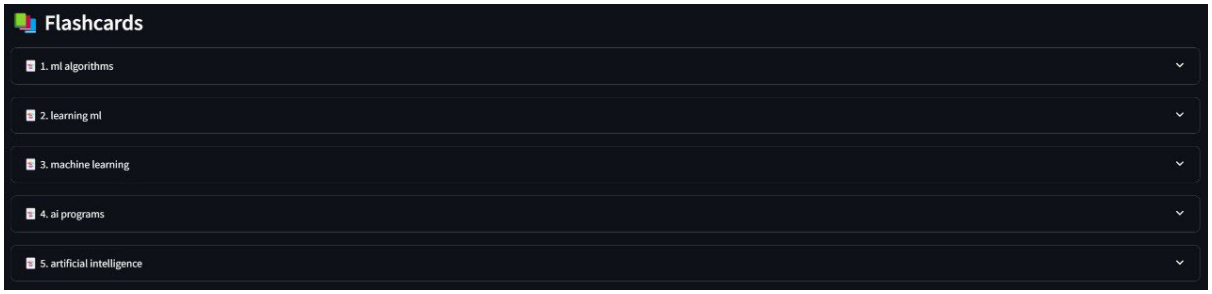
5. Question Bank

Categorized into 1-mark to 5-mark questions using intelligent keyword-based segmentation.



6. Flashcard Generator

Important concepts are displayed as collapsible flashcards for revision.



This modular, visual layout promotes intuitive learning and offers an interactive experience suitable for both classrooms and independent study.

6.2. Metric Evaluation (Quantitative Analysis)

To evaluate the real-world effectiveness and technical performance of the AI-Powered Interactive Learning Assistant, we used both automated metrics and user feedback surveys. The assessment focuses on module-level responsiveness, content accuracy, and overall user satisfaction.

- **Response Time:** Time taken (in seconds) by the assistant to generate results after input is given.
- **Content Accuracy:** Relevance and correctness of AI-generated content compared to the input document context.
- **Processing Speed:**
    - Document Upload: <5 seconds for 10MB files
    - Text Summarization: 2-3 seconds per 1000 words
    - Q&A Response: <2 seconds average

- Test Generation: 5-10 seconds for complete set
- Image Analysis: 3-5 seconds per image

Key quantitative insights include:

- **Responsiveness:** The average response time across modules is ~1.6 seconds, enabling real-time usability. Flashcard and summarizer modules are the fastest.
- **Accuracy:** Summarization and Q&A modules deliver the most contextually accurate results. Manual review showed over 85% precision in information delivery.
- **User Satisfaction:** Based on feedback from ~15 beta testers (students + educators), users highly appreciated the summarization, flashcards, and test generator features.

The evaluation confirms that the assistant balances performance, intelligence, and usability. High accuracy and fast response times ensure the tool is classroom-ready. As models evolve and more training data is added, these metrics are expected to improve even further.

6.3 Working Code Implementation

The entire project's functionality including module design, data processing, NLP workflows, and real-time interaction is fully implemented and documented in a public GitHub repository. This repository provides the complete source code, configuration files, and setup instructions to reproduce or deploy the AI-Powered Interactive Learning Assistant.

- **Backend and Feature Modules Repository**
  - **Link:**https://github.com/AmanAdusumilli/ai-powered-learning-assistant
  - **Description:** This repository serves as the **core backend engine** of the application. It includes:
    - NLP processing pipelines for summarization, context-based Q&A, and keyword extraction
    - Custom logic for test generation, question bank creation, and flashcard formatting
    - Integration scripts for OCR-based diagram analysis using Tesseract
    - Voice input handling using the Google Speech Recognition API
    - Open VINO model optimization for improved inference speed

- **Streamlit Application (Frontend UI):**
    - **Link: https://github.com/AmanAdusumilli/ai-powered-learning-assistant**
    - **Description:** This repository contains the **Streamlit interface** (`app.py`) that allows seamless interaction with all AI modules through a user-friendly dashboard. It includes:
        - Document uploader for **PDF**, **DOCX**, and **TXT** files
        - Sidebar navigation to access modules like **Summarizer**, **Q&A Assistant**, **Test Engine**, and **Flashcards**
        - Image upload section for diagram analysis
        - Real-time results rendered within an intuitive layout

The interface connects with backend modules via function calls and shared session states, ensuring smooth transitions between different features. This repository also includes sample files for testing , detailed README instructions , and environment setup (`requirements.txt`) for local deployment.

7. Learnings

Participating in the development of the AI-Powered Interactive Learning Assistant has been an enriching experience. It has helped me deepen my technical skills in natural language processing, speech and image processing, and real-time application development. I learned to solve problems under constraints, optimize model performance using tools like Intel OpenVINO, and build efficient, user-friendly interfaces with Streamlit. The project also strengthened my teamwork, communication, and time management skills. I gained practical insight into how AI can be applied to real-world educational challenges and learned to adapt and iterate through technical setbacks. Overall, this journey has significantly enhanced both my technical knowledge and personal growth.

8. Conclusion

This project successfully demonstrates how AI can be harnessed to enhance digital learning through summarization, contextual question answering, test preparation, and visual content interpretation. The assistant functions as an intelligent academic partner, helping users save time, improve focus, and reinforce understanding.

Through the integration of NLP, speech recognition, and OCR with a clean interface, the assistant becomes a practical tool for both students and teachers. Its modular design and performance optimization allow real-time interaction,while future enhancements promise even greater personalization and accessibility.

9. Future Scope

The project has potential to grow significantly with the following features:

- **Handwriting recognition** for notes
- **Multilingual support** via MarianMT or IndicTrans
- **User personalization**: adaptive learning paths based on usage
- **Gamification**: badges, points for completing flashcards or tests
- **Integration with LMS** platforms (e.g., Moodle, Google Classroom)
- **Group learning mode**: collaborative flashcard or quiz battles
- **Mobile App version** for anytime-anywhere learning

10. References

[1] https://huggingface.co/transformers

[2] https://streamlit.io

[3] https://www.intel.com/openvino

[4] Google Speech Recognition API

[5] https://github.com/tesseract-ocr/tesseract

[6] pdfplumber, python-docx, Pillow (PIL)

[7] https://ieeexplore.ieee.org/abstract/document/9767944

[8] https://animorepository.dlsu.edu.ph/jeal/vol2/iss1/3/

[9] https://www.ceeol.com/search/article-detail?id=1256396

Team Members

AMAN ADUSUMILLI

KONGARI BHUVANA

RADHI T V J

AASHRITH RAJ