

NLP Miniproject

Project Statement:

Creating multiple-choice questions (MCQs) manually is time-consuming and requires domain expertise. Automating this process using advanced AI models can assist educators and e-learning platforms by providing a scalable and efficient solution.

Project Description:

This project utilizes Google Gemini's Generative AI through its API, integrated with Langchain, to automatically generate MCQs from text data. By leveraging cutting-edge generative models, the system can create contextually accurate questions, offering one correct answer and several distractors.

Implementation:

Pre-processing Steps:

- **Input Collection:** Text data is collected from various sources, including articles or user-provided input.
- **Tokenization:** The text is broken down into tokens to extract key concepts.
- **Context Understanding:** Google Gemini's API is used to generate a high-level understanding of the text.
- **Prompt Design:** Questions are framed using Langchain, a language model orchestration tool, to ensure they follow a logical flow.
- **Distractor Generation:** The model generates incorrect yet plausible answers (distractors).

Algorithm:

- **Langchain Integration:** Handles text parsing and model communication.
- **Generative API Calls:** Google Gemini's generative model interprets the text and creates questions.
- **Post-Processing:** The system ensures MCQs are well-formatted, with one correct answer and multiple distractors.

Libraries Used:

- **Langchain:** For connecting with the language model and managing prompt design.
- **Google Gemini API:** Provides generative AI capabilities for understanding and summarizing content.
- **Pandas:** For data handling and manipulation.
- **NLTK:** For natural language processing and text preprocessing.
- **Streamlit:** For creating an interactive user interface.

Sample Input/Output:

MCQs GENERATOR

Upload Pdf or Text File

Drag and drop file here

Limit 200MB per file

Browse files

ml.txt

9.5KB

Number of MCQs

5

-

+

Insert the Subject Name

Machine Learning

16/20

Select the Difficulty of the Questions

Simple

Press Enter to submit form • 6/20

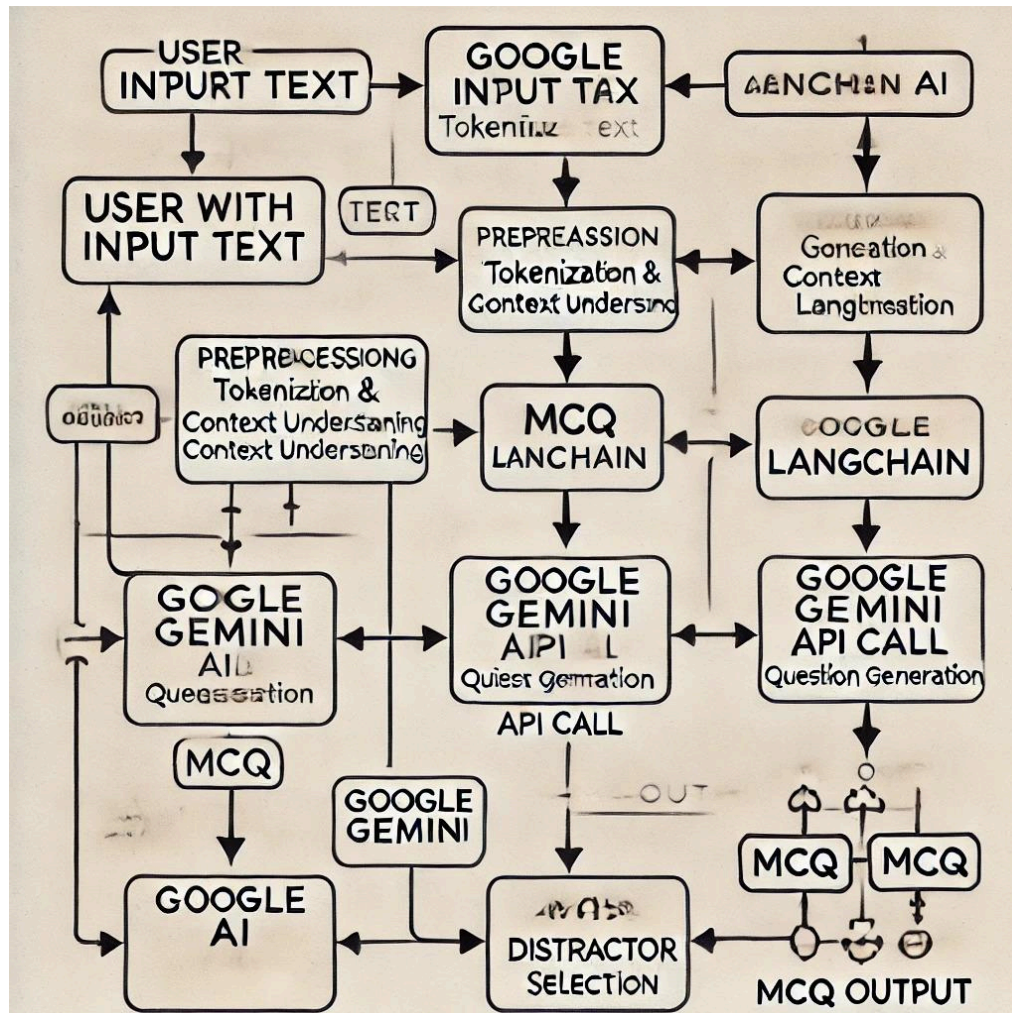
Generate MCQs

	MCQ	Choices	Correct
1	What is the main purpose of fine-tuning in deep learning?	a-> To improve accuracy on a specific task b-> To reduce the size of a model c-> To make a model more robust to noise d-> To speed up training time	a
2	Which of the following is a benefit of fine-tuning?	a-> It can improve performance on downstream tasks b-> It can be less computationally expensive than full model training c-> It can help prevent overfitting d-> All of the above	d
3	What is a common technique used to keep a model's robustness after fine-tuning?	a-> Freezing the earlier layers b-> Using dropout c-> Adding noise to the data d-> Linearly interpolating weights	d
4	Which of the following is a type of adapter used for efficient fine-tuning?	a-> LoRA b-> PEFT c-> ReFT d-> All of the above	a
5	What is a limitation of fine-tuning?	a-> It can degrade a model's robustness b-> It can be computationally expensive c-> It can lead to overfitting d-> All of the above	d

Review

Complexity Evaluation: Moderate. The questions are clear and concise, and they cover the key concepts of fine-tuning in deep learning. However, some of the questions may require students to have a strong understanding of the underlying mathematical concepts.

Flowchart:



Conclusion:

By integrating Google Gemini's Generative AI with Langchain, this project provides an automated, efficient method for generating MCQs. It enhances educational content creation by offering scalable question generation, reducing manual effort, and ensuring quality through advanced AI models.