Group members: 1. Jayamini Hewawasam (152996512)

2. Shashikala Rajapaksha (152694568)

Run the code

1. We run it in Colab.

2. Need to install necessary Python libraries.
   Before running the code, ensure you have torch and transformers installed.
   !pip install transformers faiss-cpu sentence-transformers torch datasets gradio

3. The script will download and process the e-books, extract content, and enable question-answering and summarization.

4. Here we used Gradio interface, run the corresponding cell to launch the chatbot.

Approach Overview

Problem Statement:

The project involves building a chatbot that can summarize and answer questions based on agriculture-related e-books.

1. **Data Collection (Downloading Books):**
   The script downloads agriculture-related books from Project Gutenberg. URLs of selected books are stored in a list. A loop fetches and saves each book as a .txt file.

2. **Data Cleaning and Document Processing**:
   a. The e-books are preprocessed to extract useful text.
      (remove: Headers and footers added by Project Gutenberg. Disclaimers, transcriber's notes, and metadata. Email addresses and irrelevant formatting.)
   b. The cleaned text is split into meaningful paragraphs.
   c. Short paragraphs (< 300 characters) are removed.
   d. Unwanted introductory phrases (e.g., "The Project Gutenberg", "Produced by") are filtered out.
   e. The extracted meaningful paragraphs are stored in preprocessed books.json for further use which serve as the **knowledge base** for our chatbot.

f. Text chunks are embedded using sentence-transformers.

3. **Retrieval Mechanism**:
    a. User asks a question (e.g., “How can I improve soil fertility?”).
    b. **FAISS** is used as a vector store to retrieve relevant document sections.
    c. Queries are compared against stored embeddings to fetch the most relevant context.
    d. **Uses DPR with Sentence Transformers** - It loads a Sentence Transformer model (multi-qa-mpnet-base-dot-v1), which is optimized for question-answering retrieval.

4. **Fine-tune: In step2,**
    a. Fine-tuned BART on agricultural text for better summarization quality and better question answering.

5. **Summarization & Question Answering**:
    a. **BART-based summarization model (facebook/bart-base)** handle text summarization when queries are descriptive (e.g., 'Tell me about crop rotation').
    b. **BERT-based QA models** extract specific answers for factual questions (e.g., 'What is irrigation?').

6. **User Interface**:
    a. A simple **Gradio** interface allows users to input queries and receive responses.

<u>Future Enhancements</u>

- Fine-tuning models for better domain-specific responses.
- Expanding the dataset with additional agricultural resources.
- Implementing multi-turn conversations, where the chatbot remembers previous interactions and responds contextually, to improve user engagement.
- Implement long summarization outputs as bullet points for better readability.
- Increase epochs and batch size for better learning.