**Name: Rene Etse Alordeh-Heymann**

**Roll Number : 10211100366**

**INTRO TO AI END OF SEMESTER EXAMINATION**

**1. Clear instructions on how to use each feature.**

**a) Regression**

* Go to the “Regression” tab.
* Upload a CSV dataset with numerical features and a continuous target column.
* Specify the target column name (e.g., "Price").
* The app previews the dataset and allows for basic preprocessing.
* Click “Train Model” to fit a Linear Regression model.
* View model metrics (Mean Absolute Error, R² Score) and scatter plots of predictions vs actual values.
* Input custom values to make predictions with the trained model.

**b) Clustering**

* Navigate to the “Clustering” tab.
* Upload a multi-feature dataset (e.g., customer data).
* Use the interactive slider to choose the number of clusters (k).
* The app visualizes the resulting clusters in 2D/3D (based on feature dimensionality).
* Centroids and cluster membership are displayed.
* You can download the clustered dataset.

**c) Neural Network**

* Select the “Neural Network” tab.
* Upload a classification dataset (e.g., digit recognition or binary classification).
* Specify the target column.
* Choose training parameters (epochs, learning rate).
* Train a feedforward neural network using TensorFlow.
* Visual training progress is shown (accuracy/loss graphs).
* Upload new test samples for prediction.

**d) Large Language Model (LLM)**

* Open the “LLM Q&A” tab.
* Choose one of the available datasets (e.g., 2025 Budget PDF).
* Enter your question in the provided input box.
* Receive real-time answers generated by the selected LLM model.
* Confidence score and reference text snippets are displayed alongside the answer.
* **2. For (d), a detailed description of datasets and models used [2 marks]**

Choose one dataset and one LLM approach. Example:

**Dataset Chosen**:

* 2025 Budget Statement and Economic Policy (PDF)  
  This document contains structured and unstructured text data about Ghana’s fiscal policy and government plans, ideal for natural language Q&A.

**For (d): Detailed Architecture [6 Marks]**

User Input (Q)

↓

Generate Query Embedding

↓

FAISS Vector Database

(Budget PDF Chunks)

↓

Top-K Relevant Context

↓

Prompt + Context → LLM

↓

Generated Response

**Components:**

* **Document Parsing**: PDF converted to clean text and chunked logically.
* **Embedding Model**: SentenceTransformers (all-MiniLM-L6-v2) converts chunks into vector format.
* **Vector Store**: FAISS stores the embeddings and enables fast semantic search.
* **Retriever**: Fetches the top-k most relevant text chunks.
* **LLM (Mistral)**: Takes question + context and generates accurate answers.

**Novelty Highlight**:

* Integration of open-source LLM with custom Ghanaian policy documents.
* Real-time contextual Q&A from local data sources

**For (d): Detailed Methodology [6 Marks]**

1. **Data Processing**:
   * The 2025 Budget PDF is processed using **PyMuPDF** to extract raw text.
   * Text is split into chunks (approx. 300–500 tokens) using sentence-level segmentation.
2. **Vectorization**:
   * Each chunk is embedded using SentenceTransformer.
   * Chunks are indexed using FAISS for fast nearest-neighbor searches.
3. **Retrieval**:
   * When the user asks a question, the query is embedded.
   * FAISS retrieves the top-k relevant chunks that semantically match the query.
4. **Prompt Construction**:
   * The retrieved context and the user query are formatted into a prompt.
   * Prompt is passed to the Mistral LLM using HuggingFace Transformers.
5. **Answer Generation**:
   * LLM generates a response, which is displayed to the user.
   * Optionally, metadata such as similarity scores and source snippets are shown.

**For (d): Detailed Methodology [6 Marks]**

1. **Data Processing**:
   * The 2025 Budget PDF is processed using PyMuPDF to extract raw text.
   * Text is split into chunks (approx. 300–500 tokens) using sentence-level segmentation.
2. **Vectorization**:
   * Each chunk is embedded using SentenceTransformer.
   * Chunks are indexed using FAISS for fast nearest-neighbor searches.
3. **Retrieval**:
   * When the user asks a question, the query is embedded.
   * FAISS retrieves the top-k relevant chunks that semantically match the query.
4. **Prompt Construction**:
   * The retrieved context and the user query are formatted into a prompt.
   * Prompt is passed to the Mistral LLM using HuggingFace Transformers.
5. **Answer Generation**:
   * LLM generates a response, which is displayed to the user.
   * Optionally, metadata such as similarity scores and source snippets are shown.

### For (d): Evaluate and Compare with ChatGPT [4 Marks]

|  |  |  |  |
| --- | --- | --- | --- |
| Question | ChatGPT Response | Custom LLM Response | Comparison |
| What is Ghana’s projected GDP growth in 2025? | “Ghana is projected to grow by X%” (general) | “According to the 2025 Budget, the GDP is expected to grow by 4.8%...” | Custom LLM gives exact source-based data |
| Which sectors are prioritized in the budget? | Lists general sectors | |  | | --- | |  |  |  | | --- | | Lists sectors with specific funding allocations | | Custom LLM provides more factual clarity |
| What is the total revenue projection? | Rough estimate | Extracts exact value from the document | More precise and verifiable |

**Conclusion**:

* ChatGPT provides general knowledge and lacks access to local documents.
* The custom Mistral-based LLM provides context-specific, accurate responses from Ghanaian government documents.
* Custom model shows superior performance on localized and document-grounded queries.