Modelización de problemas de la Empresa

Problem proposal

Title: Intro RAG Bake-Off for Technical Q&A

Description

Build a small Q&A system over a technical paper (in this case, a research paper from Meta) and compare how different retrieval setups help an LLM answer multiple-choice questions.

Compare **two** retrieval setups and see which one helps an LLM answer multiple-choice questions better:

- BM25 (Classic Keyword Search)
- Dense Retrieval (Embeddings)

You can use any LLM; free-tier options include the **Google Gemini** model family for generation, and **sentence-transformers** for embeddings.

Development details

We will be using a recent paper from Meta SuperIntelligence Labs, where the lab presents a new RAG framework called REFRAG. At Management Solutions, we curated a dataset from this paper with hard, multiple-choice questions to test different RAG pipelines on said questions. You will find this dataset in this file's appendix.

Dataset

The provided dataset consists of 50 multiple-choice questions. Each question includes one correct answer and three distractors. The dataset is stored in a JSON file, where each question follows the schema below:

```
{
    "question" : "What is REFRAG?",
    "answers" : {
        "A" : "A cybersecurity conference focused on reverse engineering and digital forensics.",
        "B" : "A new programming language for web development.",
        "C" : "A cloud service for data storage.",
        "D" : "A new framework developed by Meta Superintelligence Labs for RAG."
    },
    "correct_answer" : "D",
    "paper_reference" : "Meta has developed a new framework called REFRAG for RAG."
}
```

Then

1. Split the text into sensible chunks and store them in a vector database, we recommend using FAISS or ChromaDB. You can use different chunking strategies, creativity and state-of-the-art chunking will be considered towards counting your score.
2. Implement three pipelines:
    - **(A)** LLM baseline
    - **(B)** LLM + BM25
    - **(C)** LLM + Dense Retrieval
    - **(Bonus Points)** LLM + Hybrid Retrieval (A combination of BM25 and Dense Retrieval)
3. Generate answers using an LLM and save scores for multiple runs to make the LLM score statistically relevant. The LLM must provide the correct answer and the source it obtained the information from (if applicable)

4. Evaluate: **Number of correct answers** and **source attribution accuracy** (does the cited span actually contain the answer?). **Bonus points:** Add any other metrics that might be relevant.
5. Create a concise dashboard (tables/plots) comparing the baseline LLM vs. improved systems.
6. Report a short **write-up** of findings for the given pipelines.

You will be scored based on (in order of importance)

1. Reproducible experiments and clean code.
2. Solid evaluation harness and visual comparison (bar charts/tables).
3. Clear insights on when/why RAG helps, how it differs from BM25, and when to use each one — supported by evidence

Please note that state-of-the-art results are not necessary. We value careful experimental design and transparent reporting.