**PDF Chunker**

In large language models (LLMs), context length is crucial. The trend towards larger context windows has shown that LLMs struggle to maintain adequate attention on the content. To address this issue, we implement "chunking", which involves splitting a document into multiple small chunks that are retrieved during the answer generation process.

In this assignment, we need to build a chunker for the large pdf document.

**Goal:**

* Develop a chatbot that generates answers based on a given PDF document.
* Accept PDF inputs.
* Provide an API endpoint (or just the demo via command line) for posing questions and generating responses, without the need for real-time operations.

**Requirements:**

* Maximum Chunk Size: 1000 characters
* Answer Generation: When generating answers, send all related/required chunks to the LLM.
* Answer Completeness
  + Answer can span across multiple chunks, so ideal answer should list all steps/answers from the given section (it may require fetching related chunks)
* Table Queries: Capable of answering questions from tables within the LLM, allowing users to seek specific records, gather information, or make comparisons.
* **Bonus**
  + Able to generate relevant images. E.g. if a step in a pdf document has an image we should be able to send the image in the response.
  + This is different work
* Don’t use llama index and langchian - you can use the concept but not the python package from langchain / llama index. They have solved it partially but not in a great way.
* Models
  + Can use any model - ideally openai for accuracy
  + Use faiss for the vector embeddings.
  + Database - any database (we use mysql)
  + Embeddings model - you can use ada-2
  + Answer generation - GPT-4 / 4O has highest accuracy. Will recommend to use this one

**Deliverable**

* Github code
* Loom video - showcasing the approach of the problem. Please focus on strategy

**Evaluation Criteria**

* Accuracy of AI response
* High quality code
  + Modular
  + Easy to manage
* Chunking strategy with pros and cons

**Sample data**

**Some sample pdf data - in google drive**

**Same questions to test**

1. Tell me the names of all the cities where Microsoft is located? (chunk testing)
2. What’s the accuracy difference between Dexterity and mobility? (Table tests)
3. How do I include a surcharge? (For image testing - answer should have image in the answer in the order)

Reference Documents (In case you want to read)

* <https://github.com/nlmatics/llmsherpa#layoutpdfreader>
* https://github.com/Unstructured-IO/unstructured
* <https://medium.com/@salujav4/parsing-pdfs-text-image-and-tables-for-rag-based-applications-using-llamaparse-llamaindex-0f4c5ed50fb7>