RAG Mini Project Milestone #1

Summary

This assignment's goal was to prepare text data for Retrieval-Augmented Generation (RAG) by selecting and processing a set of documents, applying chunking techniques, and saving the structured text data for future retrieval.

I worked with five Word documents on the topic of Agentic AI, covering its introduction, technical aspects, applications, challenges, and future trends. The goal was to extract meaningful text from these documents, split it into structured chunks using both fixed-size chunking and semantic chunking, and save the results in a Pickle file (`chunks.pkl`) for use in Milestone 2.

The project was not just about running code; it required understanding how text data is processed, how AI systems retrieve information, and how to optimize chunking techniques for efficient retrieval.

Key Takeaways From The Project

1. **Text Preparation is Crucial for AI Models**
    a. Before AI can generate relevant answers, it must have access to well-structured, retrievable text.
        - Raw `.docx` documents contain unstructured text that is not useful for direct retrieval.
        - Extracting, cleaning, and structuring this data improves how efficiently the AI can use it.
2. **Different Chunking Methods Affect AI Retrieval**
    a. One of the most important concepts in this milestone was chunking - the process of splitting large documents into manageable and retrievable pieces.
    There were two methods:
        i. **Fixed-Size Chunking:**
            1. Breaks text into equal-length chunks (e.g., 500 characters).
            2. Simple but doesn’t preserve meaning—important concepts might be split.
        ii. **Semantic Chunking:**
            1. Uses AI embeddings to merge similar paragraphs into larger, more meaningful chunks.
            2. Preserves context and helps AI retrieve information more accurately.

Key Learning:

- Fixed-size chunking is fast but less accurate.

- Semantic chunking is smarter but computationally expensive.
- A balance between the two is needed for effective AI retrieval.

Working with AI requires a strong understanding of data pipelines. This milestone taught me that AI retrieval is not just about training a model; the quality of input data affects its performance.

I had to consider:

- How should the text be structured?
- What is the best way to split content without losing meaning?
- How can we optimize for retrieval efficiency?

This was my first exposure to data preprocessing for AI retrieval, and I now understand how RAG pipelines rely heavily on well-prepared text data.

Technical Challenges

The roadblock for me in this project was getting the right Python environment to work.

- I learned that different tools (Jupyter, Conda, pip) can install packages in different environments, which caused initial errors when trying to use `python-docx` and `sentence-transformers`.
- I solved this by checking Python versions, switching kernels, and using the correct install commands.

This Assignment helped me:

1. Understand how AI models retrieve and process text.
2. Learn the importance of chunking methods in AI-powered search.
3. Solve real-world debugging issues in Python environments.
4. Prepare clean, structured data for use in the next RAG milestone.