Semantic Search System for Policy Documents

**Overview:** This project focuses on creating a semantic search system tailored for policy documents, integrating advanced techniques for document processing, vector embeddings, and coherent answer generation. The system comprises three key layers: embedding, searching, and generation, each optimized for enhanced performance.

**Project Structure:**

1. Embedding Layer:   Explore various PDF document processing and chunking strategies.   Choose between OpenAI's embedding model or SentenceTransformers for vector representations. image
2. Search Layer:   Design three diverse queries reflecting potential user questions in policy documents.   Implement vector database searches against ChromaDB, incorporating a cache mechanism.   Enhance search results with a re-ranking block using cross-encoding models from HuggingFace. image

Rerank after cross encoding provided image

1. Generation Layer:   Design an exhaustive and instructive prompt for the Language Model (LM) to ensure coherent answer generation.Provide a few-shot example in the prompt to improve LM output.

**Performance Evaluation:** Thoroughly assess the impact of different strategies, models, and components on system performance.Consider scalability by addressing potential increases in document numbers or user queries. Getting Started:

1. Install the required libraries: pip install pdfplumber tiktoken openai chromaDB sentence-transformers -q.
2. Mount Google Drive: from google.colab import drive; drive.mount('/content/drive', force\_remount=True).
3. Follow step-by-step instructions in the codebase to implement the Embedding, Search, and Generation layers.