Documentação Funcional

Este código implementa uma interface gráfica usando **Tkinter** para processar um arquivo PDF. Ele extrai o texto do PDF, gera embeddings (representações vetoriais) para o conteúdo, realiza a classificação de temas e implementa uma função de **perguntas e respostas** que permite ao usuário fazer perguntas sobre o conteúdo do PDF.

1. Importação de Bibliotecas

O código utiliza:

Tkinter: para criar a interface gráfica.
PyMuPDF (fitz): para abrir e extrair texto de arquivos PDF.
SentenceTransformer: para gerar embeddings de sentenças usando o modelo all-MiniLM-L6-v2.
NumPy e cosine_similarity: para realizar cálculos de similaridade.
nltk: para o processamento de texto e tokenização de sentenças.

2. Configuração de Modelos e Tokenizer

Punkt: Modelo de tokenização para dividir o texto em sentenças.

all-MiniLM-L6-v2: Um modelo leve de embeddings que transforma sentenças em vetores numéricos, facilitando a análise de similaridade e a classificação de temas.

3. Classificação de Temas e Palavras-Chave

Lista de Temas: Define categorias principais como "família", "romance", "suspense", entre outros.

Dicionário de Palavras-Chave: Cada tema possui um conjunto de palavras associadas, ajudando na identificação do contexto principal do PDF com base na ocorrência dessas palavras.

4. Função de Classificação de Temas

A função de classificação usa as palavras-chave para calcular qual tema melhor representa o conteúdo do PDF. Ao encontrar palavras do dicionário dentro do texto, o tema correspondente é identificado como provável.

5. Função de Perguntas e Respostas

O código implementa uma função para responder perguntas feitas pelo usuário sobre o conteúdo do PDF. Abaixo está uma explicação detalhada:

a. Preparação do Embedding para Perguntas e Respostas

Quando o PDF é carregado, ele é dividido em frases, e um embedding é gerado para cada frase. Esse processo ajuda a transformar o conteúdo do PDF em representações vetoriais.

b. Processamento da Pergunta

Quando o usuário faz uma pergunta, o sistema:

Converte a pergunta em um vetor de embedding usando o modelo SentenceTransformer.

Calcula a similaridade entre o vetor da pergunta e os vetores de cada frase do PDF usando a similaridade de cosseno.

c. Retorno da Resposta

A frase com o valor de similaridade mais alto é selecionada como a resposta, indicando que é a mais relevante para a pergunta feita. Isso permite que o sistema forneça respostas contextuais e precisas com base no conteúdo do PDF.

6. Interface Gráfica com Tkinter

A interface gráfica permite:

Escolher o Arquivo PDF: Um botão para selecionar o arquivo PDF a ser processado.

Visualizar o Tema do PDF: Exibir a classificação do tema do texto após o processamento.

Campo de Perguntas e Respostas: Permite ao usuário digitar perguntas sobre o PDF, recebendo uma resposta baseada no conteúdo extraído.

7. Funcionalidades e Fluxo de Trabalho

Seleção do PDF: O usuário carrega o PDF, e o texto é extraído.

Classificação de Tema: As palavras-chave são utilizadas para inferir o tema geral do documento.

Perguntas e Respostas: O usuário digita uma pergunta, e o sistema retorna a resposta mais próxima com base na similaridade de cosseno.