Desafio Python — Processo Seletivo ADA

Este desafio faz parte do processo seletivo para a bolsa **Trainee LLM** do Projeto ADA – Assembly Digital Assistant, cujo foco é avaliar sua capacidade de organizar um projeto em Python, manipular arquivos, extrair informações estruturais de um PDF e integrar uma LLM local usando Hugging Face.

Você terá até o dia **11/12/2025** para desenvolver e submeter seu projeto para análise.

Objetivo do Projeto

Desenvolver uma ferramenta de linha de comando (CLI), escrita em **Python**, capaz de receber um arquivo PDF em português como entrada, extrair informações estruturais do documento e gerar um resumo do seu conteúdo usando um **modelo de linguagem local** da Hugging Face.

O projeto deve ser organizado, modular e demonstrar domínio intermediário da linguagem Python.

Funcionalidades Obrigatórias

1. Análise do PDF

A ferramenta deve ser capaz de extrair e exibir na saída padrão as seguintes informações do PDF fornecido:

- **Número total de páginas**
- **Número total de palavras**
- **Tamanho em bytes do arquivo PDF**
- **Lista das 10 palavras mais comuns** (que não sejam stopwords como a, de, o, um, uma, etc)
- **Tamanho do vocabulário** (número de palavras distintas após limpeza básica)

Essas informações devem ser processadas sem uso de IA, apenas com bibliotecas Python.

---

2. Extração de Imagens

O programa deve identificar e extrair todas as **imagens contidas no PDF**, salvando-as em um diretório `imagens/<nome-arquivo-pdf>/` dentro do projeto ou em outro local indicado pelo usuário.

Cada imagem extraída deve receber um nome único dentro do diretório indicado.

---

3. Geração de Resumo com LLM Local

Usando um **modelo local da Hugging Face** (como Gemma, Mistral, Falcon, Qwen ou equivalente), o projeto deve:

- Carregar o modelo localmente
- Gerar um **resumo textual do conteúdo do PDF**
- Imprimir o resumo na saída padrão (terminal)
- Opcionalmente salvar o resumo em um arquivo `.md` ou `.txt`

Você tem liberdade para escolher o modelo LLM, desde que ele seja executado localmente.

Funcionalidades Opcionais (valem pontos extras)

Essas funcionalidades não são obrigatórias, mas serão valorizadas:

- Detecção básica de estrutura (ex.: títulos, seções, palavras-chave)
- Suporte a PDFs grandes com tratamento de exceções
- Limpeza e normalização avançada do texto
- Uso de logs (arquivo ou console)
- Relatório final unificado em Markdown com todas as análises
- Organização de pastas bem planejada
- Tipagem com `typing`
- Testes simples

Estrutura mínima sugerida

Você pode organizar como preferir, mas esta estrutura pode ajudar:

```
src/
  pdf/
    extractor.py
    images.py
  llm/
    model.py
    summarize.py
  cli/
    arguments.py
  utils/
    text.py
    files.py
  main.py
```

Tecnologias Permitidas

- Python 3.9+
- Bibliotecas para leitura de PDF (por exemplo: `PyMuPDF`, `pypdf`, `pdfminer.six`)
- Hugging Face Transformers
- PyTorch ou equivalente para carregar o modelo
- Qualquer biblioteca padrão do Python

Não é permitido usar serviços externos ou APIs remotas.
O modelo LLM deve rodar **localmente**.

O que será avaliado

Qualidade do código

- Organização do projeto
- Clareza e legibilidade
- Nomes de arquivos, funções e variáveis
- Separação de responsabilidades

Domínio de Python

- Manipulação de arquivos
- Processamento de texto
- Estruturas de dados
- Tratamento de exceções
- Modularização
- Uso de argparse

Integração com LLM

- Carregamento correto do modelo
- Construção do prompt
- Geração consistente do resumo

Estrutura do projeto

- Pastas bem definidas
- Boa organização interna
- Evitar arquivos gigantes

Fidelidade ao escopo

- Todos os itens obrigatórios implementados
- Funcionamento correto do CLI

Funcionalidades extras (opcional)

- Cuidado com UX em linha de comando
- Códigos auxiliares bem feitos
- Logs e relatórios bem estruturados

Como enviar o projeto

1. Crie um repositório **público** no GitHub.
2. Inclua um README.md no repositório explicando:
    - Como rodar o projeto
    - Quais funcionalidades foram implementadas
    - O que você gostaria que fosse avaliado no seu projeto
3. Envie o link do repositório conforme instruções do processo seletivo.

Prazo

Você terá até o dia 11/12 para concluir o desafio e enviar o link do repositório.

Boa sorte! Queremos ver como você pensa, como organiza seu código e como resolve problemas. Não esperamos perfeição, apenas clareza, consistência e domínio dos fundamentos.