**APPLICATION BASED ON DATA EXTRACTION**

**APPROACH**:

Initially, when I read the objective, I divided the problem into two parts. The first part involved extracting data from scanned documents. After researching various tools, I decided to use PyPDF2 initially. However, upon reviewing the sample documents, I realized they were scanned documents, and using OCR (Optical Character Recognition) would likely provide more accurate results.

During my research, I discovered Tesseract, an open-source OCR engine widely recognized for its accuracy and robustness. Tesseract is developed by Google and supports over 100 languages, making it a versatile solution for text extraction from scanned documents. It utilizes machine learning algorithms to recognize characters and convert them into editable text.

Additionally, Tesseract offers various configuration options to optimize OCR results based on specific requirements. These options include language selection, image preprocessing techniques, and post-processing options to improve accuracy and handle specific document layouts or text formatting.

Then I installed tesseract from GitHub repository and explicitly added path, here is the repository link: <https://github.com/UB-Mannheim/tesseract/wiki>

PDF and Image Data Extraction with Flask

This repository contains a Flask web application that allows users to extract key-value pairs from PDF and image files using OCR (Optical Character Recognition). The application provides a user-friendly interface where users can upload a file, run the extraction process, and view the extracted data.

**Features**:

- Accepts PDF and image files for data extraction.

- Utilizes pytesseract library for OCR processing.

- Extracts key-value pairs from the uploaded files.

- Saves the extracted data into a CSV file.

- Provides a clean and responsive web interface for file upload and result display.

- Calculates processing time and word count for the extracted text.

- Includes error handling and validation for file uploads.

**Instructions**:

1. Clone the repository and navigate to the project directory.

2. Install the required dependencies using `pip install -r requirements.txt`.

3. Set the Tesseract-OCR executable path in `pytesseract.pytesseract.tesseract\_cmd`.

4. Run the Flask application using `python app.py`.

5. Access the web application in your browser at `http://localhost:5000`.

6. Upload a PDF or image file to extract key-value pairs.

7. View the extracted data, processing time, and word count on the result page.

8. The extracted data will be saved in a CSV file named 'scanned\_text.csv'.

Contributions and feedback are welcome.

ChatGPT Link: https://chat.openai.com/share/9ce7f042-da1c-4a21-9b59-b93f9381bd23

Note: This project was developed as a part of a Flask-based data extraction assignment.