AI-Based OCR System Strategy

-Sai Chiranthan H M

Objective

The aim is to develop a sophisticated OCR system capable of extracting text, tables, and images from various document types while maintaining their structural integrity. Future enhancements will leverage AI to derive insights from the extracted information.

Primary Strategy (Dependable and Scalable)

Preprocessing & Metadata Extraction

- Utilize Python libraries such as PyMuPDF or pdfminer to extract document metadata, including author details, timestamps, and embedded objects.
- Transform PDFs into images to facilitate improved processing.
- Implement layout detection tools (LayoutParser, Detectron2) to recognize text, tables, and images.

Text and Table Extraction

- Employ OCR technologies like Tesseract or PaddleOCR for text extraction.
- Utilize Named Entity Recognition (NER) with Spacy or Transformers to pinpoint essential information.
- Extract tables using tools like Tabby, Camelot, or pdfplumber, storing them in JSON format for structured access.

Image & Graph Extraction

- Identify and extract images, charts, and figures using YOLOv8 or Detectron2.

Future Enhancement – AI-Driven Insights

- Summarize text and analyze financial or structured data using large language models (LLMs) such as GPT-4 or Llama.

Secondary Strategy (Experimental)

Transformer-Based OCR & Layout Detection

- Implement cutting-edge models like LayoutLMv3 or TrOCR to enhance accuracy in document parsing.

Advanced Table Extraction

- Explore the use of Graph Neural Networks (GNN) or TableTransformer for managing complex table structures.

AI for Image and Graph Interpretation

- Utilize vision-language models such as BLIP-2 or Donut to extract insights from images and charts.

Final Deliverables

- Extract text, tables, and images from documents.
- Convert tables into structured formats like JSON.
- (Optional) Generate AI-driven insights based on the extracted data.