Document Parsing, Section Identification, and Highlighting.

1. Objective

The purpose of this project is to create a tool that allows users to upload PDF documents, intelligently detect key sections, and highlight those sections within the file. The tool enables users to select which sections to highlight and view the output directly in their browser.

Specific objectives:

- Extract structured text from uploaded PDF resumes.
- Use semantic pattern matching to detect resume sections like “Skills” or “Projects”.
- Highlight selected sections using visual annotation.
- Provide inline viewing and download options for the updated PDF.

2. Flow of Execution

Upload PDF
→ User uploads a resume file via the web interface.

Text Extraction
→ The system uses PyMuPDF to extract spans (text, font, coordinates, page).

Section Detection
→ Uses regex + keyword context to identify resume sections.

User Selection
→ Frontend shows checkboxes for selecting sections to highlight.

Highlighting
→ FastAPI applies annotations using span coordinates.

PDF Response
→ The updated PDF is streamed back for inline view/download.

3. Predefined Sections (Regex-Based)

- **Professional Summary**: “professional summary”, “career objective”
- **Name**: “name”, “full name”
- **Email**: “email pattern”
- **Phone**: “10-digit & international numbers”
- **Skills**: “skills”, “technologies”, “tools”, “frameworks”
- **Education**: “education”, “qualification”, “university”
- **Projects**: “projects”, “developed”, “built”, “implemented”
- **Certifications**: “certifications”, “certified”
- **Personal Details**: “personal information”
- **Experience**: “experience”, “employment”, “work history”

4. Features

1) PDF Text Extraction

- Extracts detailed span information: text, coordinates, font, page number.
- Converts PDFs into structured text_details objects for processing.

2) Section Detection

- Uses regular expressions to match standard headings and semantic phrases.
- Supports flexible headings like "Project Work", "Employment History", "Technologies".

3) Context-Aware Grouping

- Dynamically assigns bullet points and technical lines to their respective sections (e.g., “Skills”, “Projects”).
- Detects sections even when traditional headings are missing.

4) Highlighting Mechanism

- Applies PyMuPDF highlight_annot for visual marking.
- Rectangles are padded for clarity and grouped by selected sections.

5) User Interaction (Web Interface)

- Simple HTML form for uploading and section selection.
- Displays highlighted PDF directly in the browser.
- Supports download of the output file.

5. Technologies Learned:

PyMuPDF (Fitz): Used for reading PDF files, extracting structured text with position data, and adding annotation highlights.

Regex (re module): Employed to identify specific sections in the document by matching keywords and text patterns.

6. Conclusion

This application demonstrates how natural language and pattern recognition techniques can be applied to real-world documents like resumes. It offers a smart and simple way to enhance resume analysis by highlighting key information dynamically. The solution is user-friendly, flexible for various resume formats, and can be extended to other domains like contract parsing or academic transcript analysis.