Infosys Springboard Internship 6.0

SkillGapAI Analyzing Resume and Job Post for Skill Gap

Presented by: Ravichandra D

Under the Guidance of mentor: Praveen sir

Milestone1: AI Skill Gap Analyzer – Document Processing Pipeline

1. Introduction

The AI Skill Gap Analyzer – Document Processing Pipeline is a Streamlit-based application designed to process resumes and job descriptions. The goal of this stage (Milestone 1: Data Ingestion and Parsing) is to upload documents, extract text, clean and structure it, and prepare it for analysis.

This forms the foundation for later steps like skill comparison, gap analysis, and AI-driven recommendations.

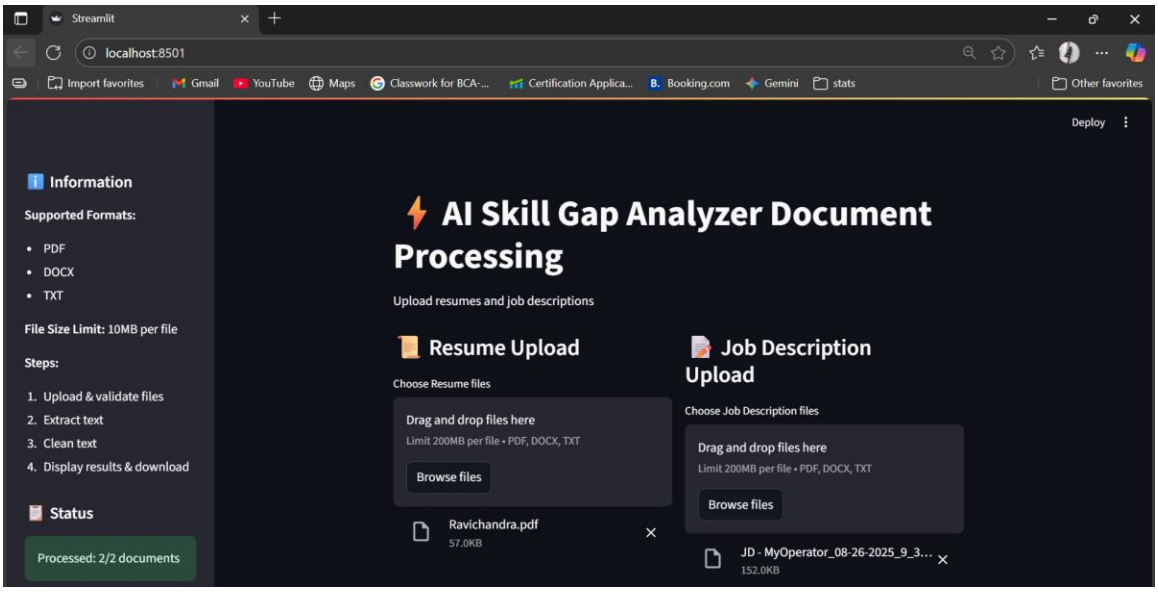
2. System Architecture

The system follows a modular architecture where each responsibility is separated into its own class and file:

Modules & Tasks

1. DocumentUploader (uploader.py) – Handles file uploads via Streamlit, supports PDF/DOCX/TXT, validates size/type.
2. TextExtractor (extractor.py) – Extracts raw text using PyPDF2, python-docx, or encoding for TXT.
3. TextCleaner (cleaner.py) – Cleans text, fixes artifacts, removes noise, and structures sections.

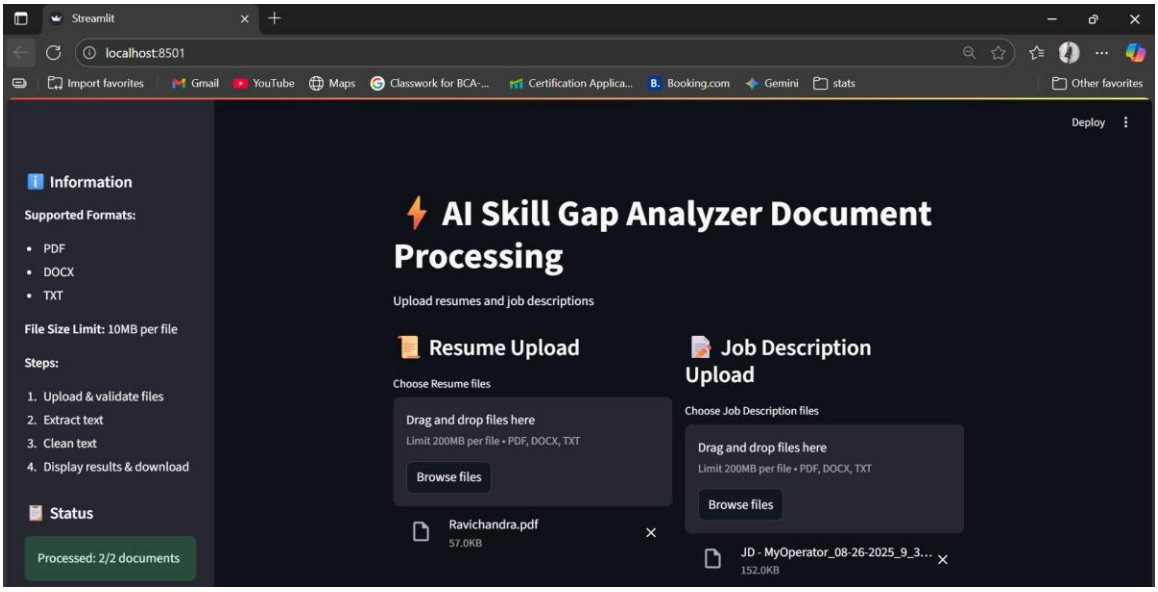
4. DocumentProcessor (processor.py) – Main pipeline that coordinates upload → extract → clean → display → download.



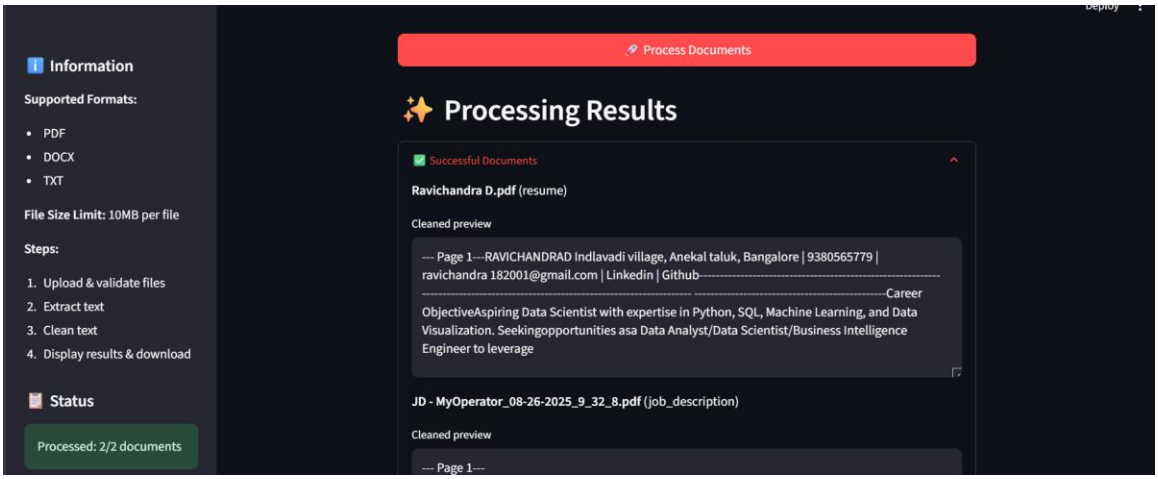
3. Workflow / Pipeline

The complete workflow consists of the following stages:

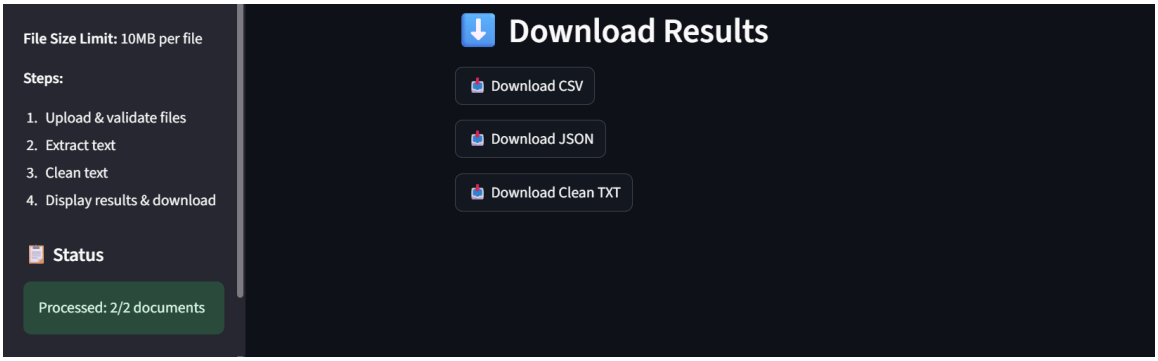
Step 1: File Upload & Validation – User uploads resumes/JDs, validated for type and size.



Step 2: Text Extraction – Extracts raw text using multiple methods for reliability.
Step 3: Text Cleaning & Structuring – Cleans extracted text, ensures readability, and organizes into sections.



Step 4: Results Display – Shows processing statistics and cleaned text preview in Streamlit.
Step 5: Export / Download – Allows download in CSV and JSON formats.



4. Features

- ✔ Multi-file upload (resumes & JDs together).
- ✔ Automatic text extraction from PDF, DOCX, TXT.
- ✔ Cleaning & preprocessing of raw extracted text.
- ✔ Section detection for structured readability.
- ✔ Progress bar & status updates during processing.
- ✔ Results preview inside Streamlit app.
- ✔ Download options: CSV and JSON.

5. Example Output

Input: Raw extracted text (from resume):

WORK EXPERIENCE
SoftwareEngineerInfosys2025Present
Python Machine Learning SQL
EDUCATION

B Tech Computer Science 2019 2022

Cleaned Output (after processing):

=== EXPERIENCE ===
Software Engineer | Infosys | 2025 - Present
Skills: Python, Machine Learning, SQL

=== EDUCATION ===
B.Tech in Computer Science (2019–2022)

6. Limitations

- Textract dependency issue: some environments face installation errors.
- Section detection uses regex → may miss unconventional headers.
- Cleaning may sometimes merge unrelated lines if document formatting is poor.
- Large PDFs (>10MB or scanned images) are not supported.

7. Future Improvements

- Support for scanned PDF OCR (via pytesseract).
- More robust section detection with ML/NLP models.
- Add similarity analysis between resume & job description.
- Improve UI with search and highlight functionality.
- Support for additional export formats (Excel, Markdown).

8. Conclusion

This pipeline successfully addresses Milestone 1: Data Ingestion and Parsing of the AI Skill Gap Analyzer. It provides a reliable way to upload documents, extract and clean text, organize content into structured sections, and export results for further analysis.

It lays the groundwork for skill extraction, gap detection, and recommendation generation in the later milestones.