Skill Gap Analyzer

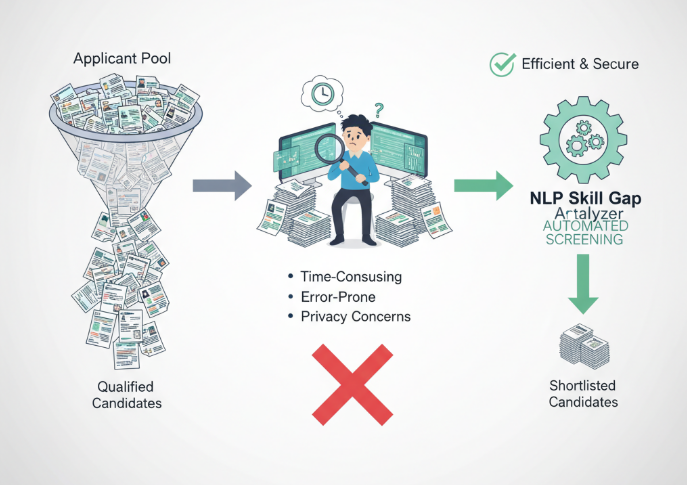
Team Member: Srihitha K

Milestone: 1 - Data Processing Pipeline and Extraction Baseline

Date: [Insert Date of Submission]

# 1. Introduction & Problem Statement

In today’s competitive job market, recruiters and hiring managers face significant challenges due to the sheer number of applications received per job opening. The manual screening of resumes against Job Descriptions (JDs) is:  
- Time-consuming – reviewing many resumes creates bottlenecks.  
- Error-prone – unstructured formats (e.g., different section headers) complicate comparison.  
- Risky for privacy – resumes contain Personal Identifiable Information (PII).  
  
**Objective:** This project aims to automate the initial screening step through a Skill Gap Analyzer using Natural Language Processing (NLP).



# 2. Objectives of Milestone 1

The goal of Milestone 1 was to establish a secure and robust data processing pipeline to prepare standardized input for later skill gap analysis.

Key Achievements:

- **Pipeline Development**: End-to-end pipeline for ingesting, cleaning, and extracting skills.  
**- Application Prototype**: Streamlit-based web app for interactive demonstration.  
- **Robustness & Privacy**: Multi-format support (PDF, DOCX) and PII redaction.

# 

# 3. Solution Architecture & Technology Choices

System Architecture: A modular, four-stage linear pipeline ensures clarity and scalability:

Document Ingestion → Preprocessing/Cleaning → Skill Extraction (Rule-Based) → Analysis/Output (Streamlit)

## Technology Stack

|  |  |  |
| --- | --- | --- |
| Component | Technology | Justification |
| Web Application | Streamlit | Rapid prototyping; easy interactive UI with Python. |
| File Ingestion | pdfplumber, python-docx | Reliable text extraction from resumes (PDF, DOCX). |
| Core NLP | spaCy | Industrial-strength NLP, optimized for rule-based phrase matching. |
| Skill Matching | spaCy PhraseMatcher | Prioritizes precision using exact matches. |
| Privacy/Cleaning | Regex | Robust detection and redaction of sensitive PII. |

# 4. Technical Implementation: The Pipeline

## 4.1 Document Ingestion

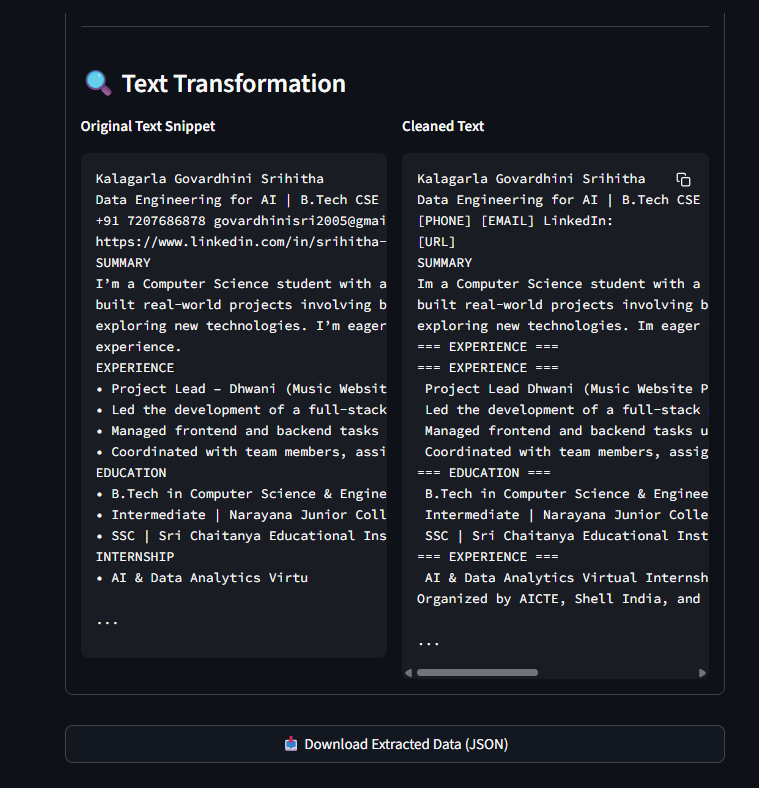
Handles .pdf, .docx, and .txt formats using specialized readers.

## 4.2 Preprocessing & Cleaning

- PII Redaction: Emails, phone numbers, and dates replaced with placeholders.  
- Section Normalization: Maps variable headers into standardized tags.

## 4.3 Skill Extraction

Skills are extracted via a master dictionary using spaCy’s PhraseMatcher. The output is a unique, sorted list of skills.



# 5. Demonstration and Results

Streamlit Application Features:  
- Upload multiple resumes + one JD.  
- Cleaned Text View – side-by-side before/after redaction.  
- Extracted Skills View – unique skill list.  
- Analytics Dashboard – bar chart of skill frequency.



## Advantages & Metrics

|  |  |
| --- | --- |
| Advantage | Description |
| Automation | Reduces manual resume screening effort. |
| Standardization | Converts messy resumes into structured, machine-readable format. |
| Privacy | PII redacted automatically, ensuring compliance. |

Performance Metrics: Total Documents Processed, Unique Skills Identified, Average Text Optimization %

# 6. Challenges Overcome

- File Diversity: Reliable parsing for PDFs and DOCX.  
- Data Quality & Privacy: Regex-based sanitization ensures clean, secure inputs.  
- Header Inconsistency: Rule-based normalization applied.

# 7. Conclusion & Future Work

Milestone 1 successfully established the foundation for the Skill Gap Analyzer: ingestion, cleaning, and rule-based skill extraction.