“AI-Powered Resume Ranker” By Lakshmi Mani Sankar Danda

--

1. Introduction [Internship Project – Elevate Labs (AIML Domain)]:

In the modern hiring landscape, organizations often receive hundreds of resumes for a single job profile, making it challenging for HR teams to manually screen all candidates. To address this issue, we developed an AI-Powered Resume Ranker – a smart system that uses Natural Language Processing (NLP) to automatically evaluate and rank resumes based on their relevance to a given job description.

What sets this system apart is its ability to **understand language contextually**, matching skills, experience, and roles from resumes to the needs of the job posting. Built using cutting-edge NLP tools like SpaCy and machine learning techniques like TF-IDF vectorization, the system offers both speed and accuracy, making it a valuable tool for modern, data-driven HR teams. Rather than manually scanning through documents, recruiters can now upload resumes and instantly get a sorted list of the most suitable candidates.

2. Abstract:

This project leverages NLP techniques to automate the resume shortlisting process. The system extracts and processes resume content, compares it against a job description, and assigns a relevance score to each resume. Using a Flask web application, recruiters can upload resumes, provide job descriptions, and instantly receive ranked results with downloadable reports. The project aims to reduce manual effort and improve the efficiency of the recruitment process.

3. Tools Used:

- Programming Language: Python
- NLP Library: SpaCy
- Vectorization: Scikit-learn (TF-IDF)
- Web Framework: Flask

- PDF Processing: PyMuPDF / pdfminer / pdfplumber (any one)
- Front-End: HTML, CSS (optional Bootstrap for styling)

4. Steps Involved in Building the Project:

1. **Resume Text Extraction:** Extract raw text from PDF resumes using PyMuPDF or similar libraries.
2. **Text Preprocessing:** Clean and tokenize text using SpaCy (removal of stopwords, punctuation, lemmatization).
3. **Job Description Input:** Take job description as input from the recruiter via a web form.
4. **Vectorization:** Apply TF-IDF (Term Frequency–Inverse Document Frequency) to convert resumes and job description into numerical vectors.
5. **Similarity Scoring:** Use cosine similarity to compute how closely each resume matches the job description.
6. **Ranking Logic:** Rank resumes based on their similarity score in descending order.
7. **Flask Web Application:** Build a user-friendly web UI to upload resumes and input job descriptions.
8. **Result Display & Report Generation:** Show ranked candidates in the browser and provide an option to download a report (CSV or PDF) for HR usage.

5. Conclusion:

The AI-Powered Resume Ranker successfully automates the tedious process of manual resume screening. It helps HR professionals save time by instantly identifying top candidates aligned with job requirements. With NLP at its core and a user-friendly web interface, this solution is scalable, efficient, and practical for real-world hiring processes. This project demonstrates the potential of AI and machine learning in revolutionizing traditional recruitment workflows.

My GitHub Repository [DandaLakshmiManiSankar] : AI-Powered Resume Ranker