Resume Parser and Job Description Matcher

Approach to the Task

Data Collection and Preparation:

The data was in a folder which contained multiple files and each files contained Several PDF files of Resumes. I used libraries like pdfplumber to extract informations like Category(Job Role), Skills, Education and Experience from the resume and cleaned it from unwanted characters.

The job descriptions were stored in a Dictionary format with relevant information such as job titles and descriptions, I converted it into a dataframe inorder to make handling

easy

Embedding Text Data:

I used a pre-trained language model distilBERT to convert the cleaned text data into numerical embeddings. This allowed for meaningful comparisons between job descriptions and resumes.

Calculating Similarity:

Cosine similarity was used to calculate how similar each candidate's resume was to each job description.

Higher similarity scores indicated a better match between the candidate and the job.

Challenges faced and solutions:

Text Data Extraction:

Resumes and job descriptions can vary widely in length and format.

Extracting relevant information with precision is a major challenge

Solution: I used advanced libraries like pdfplumber to extract relevant informations and employed text cleaning to standardize the text data, removing irrelevant characters.

Large-Scale Processing:

Processing a large number of job descriptions and resumes can be computationally intensive.

Solution: I processed data in batches to manage memory and computational resources effectively.

Choosing an Embedding Model:

Selecting the right pre-trained language model is crucial for accurate embeddings.

Solution: I used a widely accepted model like distilBERT, but other models can also be considered depending on specific requirements.

**Recommendations and Insights:**

1. **Ranking Candidates:** The similarity scores provide a ranking of candidates for each job description. Employers can focus on the top-ranked candidates for further evaluation.
2. **Customization:** The approach allows for customization. You can fine-tune the model or use different language models to better align with specific industries or job roles.
3. **Efficiency:** Batch processing helps manage large datasets efficiently, making it suitable for handling substantial volumes of job descriptions and resumes.
4. **Human Review:** While similarity scores are helpful, they should be used as a screening tool. Human review is essential to understand the context and qualifications of candidates fully.
5. **Feedback Loop:** Continuously updating and refining the model with feedback from the hiring process can improve its accuracy over time. This approach streamlines the candidate matching process, making it more efficient and data-driven. However, it should complement, not replace, human decision-making in the hiring process.