Project Title: Resume Matching Tool for Job Descriptions

Objective:

The goal of this project is to develop a small AI/ML-based tool that takes a job description and a folder containing resumes as input and produces a matching percentage for each resume in terms of relevance to the job description. This tool will help streamline the hiring process by automating the initial screening of resumes.

Duration:

1 to 2 weeks

Functional Requirements:

Input Parameters:
- Job Description: A text document or string containing the details of the job requirements.
- Resumes Folder: A folder containing individual resume files (e.g., in PDF or text format).

Text Preprocessing:
- Implement a text preprocessing module to clean and normalise both the job description and resumes.
- Tokenization, lowercasing, and removal of stop words are suggested.

Feature Extraction:
- Use Natural Language Processing (NLP) techniques to extract relevant features from the job description and each resume.
- Consider methods like TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings.

Matching Algorithm:
- Develop a matching algorithm that calculates the similarity or relevance between the job description and each resume.
- Common similarity metrics like cosine similarity or Jaccard similarity may be considered.

Output:
- Generate a matching percentage for each resume, indicating the degree of similarity to the job description.
- Display or store the results in a user-friendly format.

User Interface (Optional):
- If time permits, create a simple user interface for easy interaction.
- Allow users to input the job description, select the resumes folder, and view the matching results.

Technical Stack:

- Python (preferred language)
- Libraries: NLTK, spaCy, scikit-learn (or other relevant libraries)
- Use any suitable machine learning framework for model development (e.g., scikit-learn, TensorFlow, PyTorch)

Testing:

- Develop unit tests to ensure the correctness of individual components.
- Perform end-to-end testing using sample job descriptions and resumes.

Documentation:

- Provide clear and concise documentation for the codebase, including instructions for setting up the environment and running the tool.
- Document the approach used for text preprocessing, feature extraction, and the matching algorithm.

Delivery:

- Submit the codebase with relevant documentation.
- Provide a brief demo or walkthrough to explain how to use the tool.

Note:

This project is designed to be a learning experience for interns, focusing on fundamental concepts in NLP and machine learning. Encourage interns to explore and discuss alternative approaches, and provide guidance as needed.