Resume Evaluator: A Machine Learning-Based Candidate Ranking System

Team Members:

- Fawad Mughal - BSDSF22M036
- Subhan Ali - BSDSF22M030
- Muhammad Bilal - BSDSF22M024

Semester: 6

Course: Data Mining and Machine Learning

Supervisor: Dr. Adnan Abid

1. Problem Statement

Hiring the right candidate is a critical task for any organization, often beginning with manually screening hundreds of resumes per job opening. This process is time-consuming, prone to bias, and lacks scalability. Our project aims to automate the initial resume screening stage by leveraging machine learning to evaluate and rank candidate resumes based on how well they match a given job description.

2. Objectives

- To develop a machine learning-based system that matches resumes to job descriptions.
- To rank candidates based on relevance, skills, and experience.
- To provide HR personnel with a user-friendly React frontend to upload job descriptions and resumes.
- To ensure transparency and interpretability in how candidates are ranked.

3. Proposed Methodology

3.1 Data Preprocessing

- Extract text from resumes (PDF/DOCX) and job descriptions.
- Clean and tokenize text using NLP techniques (e.g., stopword removal, lemmatization).
- Generate dense embeddings using **SBERT** for semantic similarity computation.

3.2 Feature Engineering

- Extract structured features: years of experience, skills, education level.

- Use keyword similarity and semantic matching for scoring.

3.3 ML Models
- We'll use a pretrained **Siamese BERT model** to encode both the job description and resumes into dense semantic vectors.
- Cosine similarity between these embeddings will provide a base **semantic relevance score**.

3.4 Frontend and Integration
- React frontend to upload job descriptions and resumes.
- Backend API (Flask) to handle model inference and ranking.
- Display ranked candidates with relevance scores and key highlights.

4. Dataset Description
- **Primary Sources:** Public resume datasets (e.g., Kaggle's Resume Dataset).
- **Format:** PDF/DOCX for resumes; text or JSON for job descriptions.
- **Size:** Approx. 1,000–5,000 labeled entries (if available); else unsupervised ranking.

5. Expected Outcomes

- Customizable ranking system that adapts to HR priorities via dynamic weight adjustment.
- A responsive web interface that allows HR users to upload data and view ranked candidates.
- Insights dashboard showing key matching attributes and potential skill gaps.

6. Feasibility & Originality
While resume parsing tools exist, our project's originality lies in integrating a ranking algorithm tailored to job descriptions with a complete frontend solution for HR professionals considering the weightage of fields.