HireIntel → An AI-Powered Resume Ranker

🧠 **Using NLP & Semantic Similarity for Smart Resume Matching**

---

1. Introduction

In the modern hiring landscape, employers often receive hundreds of resumes for a single job posting, making manual shortlisting time-consuming and error-prone. This project introduces an intelligent solution — a resume ranking system that uses AI and Natural Language Processing (NLP) to compare multiple resumes against job descriptions (JDs) and rank them based on relevance.

---

2. Abstract

This project leverages transformer-based sentence embeddings and cosine similarity to score and rank resumes based on how closely they match the content of job descriptions. It supports .pdf and .docx files, auto-detects which files are JDs vs. resumes, and provides a sorted output along with a visual bar chart and downloadable results.

---

3. Tools Used

- **Python**: Core programming language
- **Gradio**: For building the interactive UI
- **PDFPlumber / python-docx**: Text extraction from files
- **Sentence-Transformers**: Semantic embeddings using all-MiniLM-L6-v2
- **Scikit-learn**: Cosine similarity for scoring
- **Matplotlib / Pandas**: Data visualization and tabulation
- **TF-IDF (TfidfVectorizer)**: Extracting key differentiators

- **Tempfile / OS**: Temporary file storage and cross-platform path handling

---

4. Steps Involved in Building the Project

a) File Handling

Users upload multiple .pdf or .docx files. The program reads each file and extracts raw text using pdfplumber or python-docx.

b) Classification

Each document is classified as either a **resume** or **JD** using keyword heuristics and filename hints. Keywords like “experience”, “education” suggest resumes, while “responsibilities”, “qualifications” suggest JDs.

c) Semantic Scoring

Each resume and JD is encoded using the all-MiniLM-L6-v2 model from Sentence Transformers. Cosine similarity is calculated between embeddings to produce a match score.

d) Keyword Extraction

TF-IDF is used to extract the top 5 discriminative keywords between each resume and JD pair.

e) Visualization

A horizontal bar chart shows resume match scores. A CSV file of the ranked results is also generated for download.

---

5. Conclusion

The AI-Powered Resume Ranker effectively reduces the effort required in shortlisting candidates. It combines the power of NLP and machine learning to deliver fast, objective, and accurate results. The system can be a valuable tool for HR departments and recruitment agencies.