Resume Matching System Using NLP for Personio

Author: Venkata Sandeep Kumar Reddy

Date: May 26, 2025

A real-world NLP project using SBERT & cosine similarity to score resume-job fit.

Abstract

This project focuses on building a Resume Matching System inspired by Personio, a leading HR tech company. By applying Natural Language Processing (NLP) techniques, it automates the comparison of resumes against job descriptions, scoring how well a candidate matches a role. We use Sentence-BERT for embeddings and cosine similarity for scoring.

Problem Statement

Recruiters and hiring managers often spend excessive time manually scanning resumes. This process is not only time-consuming but also inconsistent and prone to bias. The goal is to build a system that can objectively match resumes to job descriptions and return a similarity score to aid decision-making.

Use Case: Personio

Personio is a German HR software company offering an applicant tracking system. A resume matching tool aligns with Personio's mission to streamline hiring by adding intelligent filtering.

Data Source and Description

We used a real resume uploaded in PDF format and a realistic job description for a Data Scientist role at Personio. The resume was parsed using PyMuPDF and the JD was embedded directly in code.

PDF Text Extraction

The resume PDF was parsed using the PyMuPDF library. Text was extracted page by page and cleaned. This allowed us to preprocess the resume text into a format suitable for NLP vectorization.

Text Vectorization (Sentence-BERT)

We used the pre-trained 'all-MiniLM-L6-v2' Sentence-BERT model to convert both resume and job description text into semantic embeddings. These embeddings capture contextual meaning better than traditional TF-IDF.

Similarity Scoring (Cosine Similarity)

The cosine similarity between the SBERT embeddings was calculated using sklearn. This results in a score between 0 and 1, indicating the level of match. The result for this project was 0.47.

Model Evaluation

The resulting score of 0.47 suggests a weak match. This indicates the resume lacks sufficient alignment with the job description. This feedback can be used to tailor the resume more closely to target roles.

Result Interpretation

A score below 0.50 is considered a weak match. A score between 0.50-0.75 indicates moderate alignment, while a score above 0.75 is a strong match. This resume would benefit from highlighting NLP, HR analytics, and deployment skills more prominently.

Benefits for HR Teams

This tool can save recruiters time by automatically filtering top-matching resumes. It adds consistency and reduces bias in resume screening, especially for high-volume hiring.

Limitations

The system currently scores only one resume against one job description. It doesn't account for formatting, visual design, or soft skills. It also assumes both texts are well-written and clean.

Future Work

Future versions can support batch processing of resumes and job descriptions, visual match breakdowns (e.g., which sections matched), and integration into ATS platforms like Personio.

Deployment Plan

The model can be deployed as a Streamlit web app where HR managers can upload resumes and job descriptions. The app would return a match score and interpretation instantly.

Conclusion

This project demonstrates the real-world application of NLP in HR tech. It provides measurable, actionable results that can improve hiring decisions. It also showcases how job seekers can benchmark their resume fit to target roles.

Appendix: Sample Code

```
model = SentenceTransformer('all-MiniLM-L6-v2')

resume_embedding = model.encode(resume_text, convert_to_tensor=True)

jd_embedding = model.encode(job_description, convert_to_tensor=True)

from sklearn.metrics.pairwise import cosine_similarity

similarity_score = cosine_similarity(

    [resume_embedding.cpu().numpy()],

    [jd_embedding.cpu().numpy()]

)[0][0]
```