AI-powered Resume Screening and Ranking System

A Project Report

submitted in partial fulfillment of the requirements

of

AICTE Internship on AI: Transformative Learning

with

TechSaksham – A joint CSR initiative of Microsoft & SAP

by

Shaik Sohin, sohindillu@gmail.com

Under the Guidance of

Mr. Saomya Chaudhary

ACKNOWLEDGEMENT

I would like to take this opportunity to express my sincere gratitude to my mentor, **Mr. Soumya Chaudhary**, for his unwavering support, expert guidance, and valuable insights throughout the course of this internship. His encouragement and constructive feedback have been instrumental in refining my approach and deepening my understanding of AI.

This internship provided me with a fantastic opportunity to apply my knowledge in a real-world context, strengthening my technical and problem-solving abilities. The experience of working independently on this project has been both challenging and rewarding, offering me a more comprehensive understanding of AI-based projects.

I also wish to acknowledge the organization for offering me this internship, which has played a significant role in my growth and has helped me enhance both my academic knowledge and practical expertise in AI-driven applications.

Name : Shaik Sohin

Email : sohindillu@gmail.com

ABSTRACT

The recruitment process often involves extensive manual effort in screening resumes, making it time-consuming and inefficient. This project introduces an AI-powered Resume Screening and Candidate Ranking System to streamline the hiring process. Using Natural Language Processing (NLP) and Machine Learning techniques, the system compares resumes with job descriptions and ranks candidates based on relevance using *cosine similarity*. Implemented with Python, Streamlit, and Scikit-learn, it automates resume evaluation, reducing bias and improving hiring efficiency. This report outlines the system's methodology, implementation, and potential enhancements for future improvements.

TABLE OF CONTENT

Abstract..**I**

Chapter 1. Introduction..**1**

1.1 Problem Statement..1

1.2 Motivation..1

1.3 Objectives 2

1.4. Scope of the Project 2

Chapter 2. Literature Survey **4**

Chapter 3. Proposed Methodology..**7**

Chapter 4. Implementation and Results..**9**

Chapter 5. Discussion and Conclusion..**11**

References...12

LIST OF FIGURES

Figure No.	Figure Caption	Page No.
Figure 1	System Architecture of Resume Screening System	**7**
Figure 2	Job Description Input and Resume Upload Interface	**9**
Figure 3	Ranked Resume Output with Similarity Scores	**10**

CHAPTER 1

Introduction

1.1 Problem Statement:

Recruiters and hiring managers face the challenge of manually screening a large number of resumes, making the hiring process inefficient, time-consuming, and prone to human biases. The traditional methods of shortlisting candidates often lead to inconsistencies and may overlook suitable applicants. Automating the resume screening process using AI can significantly improve efficiency, accuracy, and fairness in candidate evaluation.

1.2 Motivation:

With the increasing number of job applicants, organizations need a fast and reliable way to assess resumes without compromising quality. AI-driven resume screening reduces human workload, eliminates bias, and enhances decision-making by ranking candidates based on job description relevance. This project aims to address these challenges by leveraging NLP and machine learning techniques.

1.3 Objective:

The primary objectives of this project are:

- To develop an AI-based resume screening system that ranks candidates based on job description relevance.
- To implement NLP techniques such as TF-IDF vectorization and cosine similarity for textual analysis.
- To create a user-friendly interface using Streamlit for easy job description input and resume uploads.
- To improve recruitment efficiency by automating the initial screening process.

1.4 Scope of the Project:

The AI-powered Resume Screening and Candidate Ranking System is designed to enhance the hiring process by automating resume evaluation. This project utilizes Natural Language

Processing (NLP) techniques to analyze resumes against job descriptions, enabling efficient and accurate candidate ranking.

In-Scope:

To ensure a comprehensive and efficient approach to AI-driven resume screening, this project includes the following key aspects:

1. **Resume Text Extraction**: Extracting textual content from PDF resumes using PDFPlumber.
2. **OCR Support**: Supporting text extraction from scanned PDFs to enhance document processing capabilities.
3. **NLP-Based Analysis**: Implementing cosine similarity and TF-IDF vectorization to process job descriptions and resumes.
4. **Candidate Ranking**: Using similarity measures to rank resumes based on relevance.
5. **Cut-off Feature**: Allowing recruiters to set thresholds for selecting top candidates.
6. **Automated Processing**: Eliminating manual screening by providing a ranked list of candidates.
7. **User Interface & Deployment**: Developing an interactive dashboard using Streamlit for recruiters to upload resumes and view rankings.

Out-of-Scope:

1. **Context-Aware Resume Understanding –** The system does not use deep learning models or word embeddings (such as BERT) for semantic analysis; it relies on cosine similarity and TF-IDF.
2. **Multilingual Resume Processing –** Currently, the system only supports resumes written in English.
3. **Grammar and Spelling Correction –** The system does not assess grammar quality or correct spelling errors in resumes.
4. **Advanced Resume Formatting Analysis –** While it extracts text from PDFs, it does not analyze resume structure, design, or layout.

5. **Soft Skill Assessment –** The model evaluates textual content only and does not analyze interpersonal skills, communication, or personality traits.

This scope ensures that the project remains focused on its core goal of automating and optimizing the resume screening process. Future enhancements may include advanced semantic understanding, multilingual support, and deeper candidate profiling to further improve the system's capabilities.

CHAPTER 2

Literature Survey

2.1 Literature Review

Traditional Applicant Tracking Systems (ATS) primarily rely on keyword matching, which often leads to inaccuracies in candidate evaluation due to a lack of contextual understanding. To address these shortcomings, AI-driven approaches such as TF-IDF, Word2Vec, BERT, and transformer-based architectures have been introduced. Large Language Models (LLMs) like GPT and Gemini further enhance resume screening by understanding complex sentence structures and contextual relevance. These advancements allow for a more refined and accurate ranking process. In this project, we leverage NLP techniques, including TF-IDF, cosine similarity, and vectorization, to improve resume-job description matching, ensuring a more efficient and unbiased hiring process.

2.2 Existing Models, Techniques, and Methodologies

1. **NLP-Based Techniques** (TF-IDF & Cosine Similarity):
    - TF-IDF (Term Frequency-Inverse Document Frequency): Used for ranking resumes by assessing the importance of words relative to the job description.
    - Cosine Similarity: Measures the similarity between resumes and job descriptions by comparing their vectorized representations.

2. **Word Embedding Models** (Word2Vec):
    - Converts words into vector representations, allowing semantic similarity comparisons rather than just keyword matching.

3. **Transformer-Based Models** (BERT, RoBERTa, XLNet):
    - BERT (Bidirectional Encoder Representations from Transformers): A deep learning model that understands context by analyzing words bidirectionally.
    - RoBERTa, XLNet: Improvements over BERT that refine training techniques and enhance performance.

4. **Large Language Models** (LLMs – GPT, Gemini):
   - LLMs are advanced AI models built on transformer architectures.
   - Models like GPT (Generative Pre-trained Transformer) and Gemini can understand complex sentence structures, relationships, and intent within resumes, making them more effective for screening.

2.3 Gaps in Existing Solutions and Project Contributions

Traditional resume screening systems and AI-based models have made significant advancements, yet they still present notable limitations:

Gaps in Existing Solutions:

1. **Keyword-Based Limitations:** Traditional ATS heavily rely on keyword matching, often leading to false positives or false negatives due to a lack of contextual understanding.
2. **Limited Context Awareness:** Many AI models struggle with understanding the deeper meaning behind resumes and job descriptions, failing to capture nuanced relationships.
3. **Bias in Screening:** Pretrained models may inadvertently carry biases from training data, impacting fair candidate evaluation.
4. **Lack of Ranking Mechanisms:** Most existing systems focus on binary filtering (match/no match) instead of ranking candidates based on relevance.

How This Project Addresses These Gaps:

- **Improved Candidate Ranking**: Instead of simple filtering, the system ranks resumes based on TF-IDF vectorization and cosine similarity, ensuring more accurate shortlisting.
- **Reduced Screening Bias**: By automating the evaluation process and standardizing ranking criteria, the system minimizes human biases in shortlisting candidates.
- **Integration of Advanced NLP Techniques**: The project applies TF-IDF, cosine similarity, and vectorization to improve accuracy in resume-job description matching.

- **Cutoff-Based Filtering**: The system includes a configurable cutoff mechanism, allowing recruiters to filter out lower-ranked resumes efficiently.

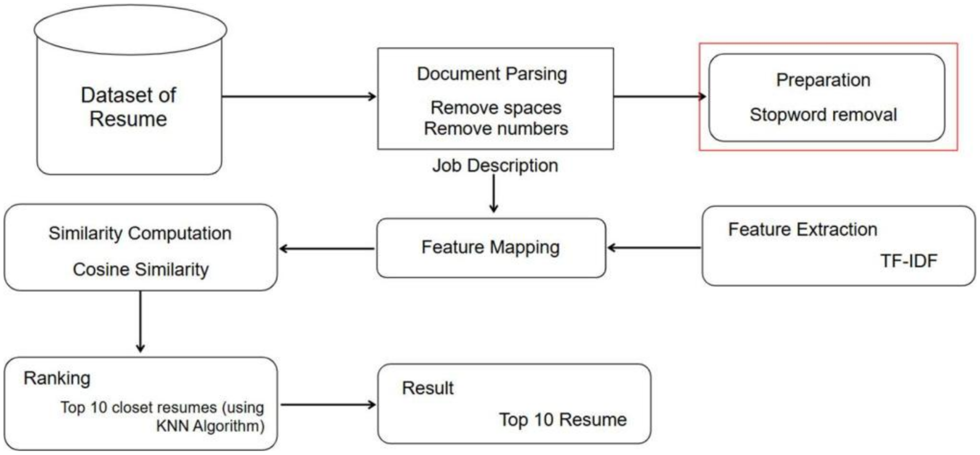
This approach enhances the hiring process by ensuring faster, more reliable, and unbiased resume evaluation. By reducing manual effort and minimizing errors, it streamlines candidate screening, leading to more efficient and data-driven recruitment decisions.

CHAPTER 3

Proposed Methodology

3.1 System Design

The proposed system automates resume screening by extracting and analyzing textual features from resumes and comparing them against the job description. It processes the documents, computes similarity scores, and ranks candidates based on relevance. The system helps recruiters by allowing them to select the top N candidates, making the shortlisting process more efficient and accurate.



Explanation of the Diagram:

1. **Dataset of Resume**: The system takes a dataset of resumes as input, typically in text or PDF format.
2. **Document Parsing and Preprocessing**: This step involves cleaning the text by removing spaces, numbers, and stopwords to refine the textual content.
3. **Feature Extraction** (TF-IDF): The processed resumes are transformed into numerical representations using the Term Frequency-Inverse Document Frequency (TF-IDF) technique.
4. **Feature Mapping**: The extracted features from resumes are mapped against the job description to establish a comparison basis.

5. **Similarity Computation**: Cosine similarity is used to measure how closely a resume matches the job description.
6. **Ranking**: The system provides flexibility by allowing users to either select the top N candidates based on relevance.
7. **Result**: The system outputs a refined list of the most suitable candidates, aiding recruiters in making more efficient hiring decisions.

3.2 Requirement Specification

To implement this project, the following tools and technologies are required:

3.2.1 Hardware Requirements

- **Processor:** Intel Core i5 (8th Gen) / AMD Ryzen 5
- **RAM:** 8GB
- **Storage:** 10GB free space
- **GPU (Optional):** Not required (unless scaling for deep learning models)
- **Internet Connection**: Required for deployment and external library installations

3.2.2 Software Requirements

- **OS:** Windows 10/11, Ubuntu 20.04+, MacOS
- **Programming Language:** Python 3.8+
- **Libraries & Frameworks:**
  - **Resume Parsing:** PyPDF2 (for extracting text from PDFs)
  - **Data Handling:** Pandas (for managing extracted resume data)
  - **Feature Extraction:** TfidfVectorizer (for text vectorization)
  - **Similarity Computation:** Cosine Similarity (for ranking resumes based on job descriptions)
  - **Deployment & UI: Streamlit (for creating an interactive interface)**
- **IDE:** VS Code, Jupyter Notebook

These specifications ensure efficient execution and scalability of the resume screening system. With the recommended setup, users can seamlessly process and rank resumes with optimal performance.

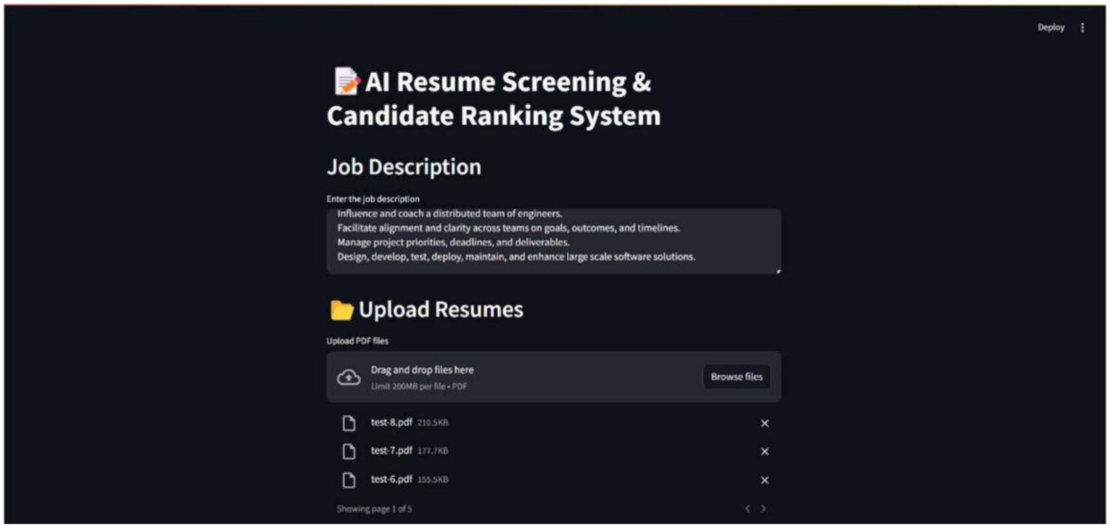
CHAPTER 4

Implementation and Result

4.1 Snapshots of Results

In this section, we will walk through the step-by-step process of the resume screening and ranking system using snapshots of the application interface. Each step will demonstrate a key stage, from entering the job description to obtaining ranked candidates. These snapshots will provide a clear visual representation of how resumes are uploaded, processed, and evaluated based on their relevance to the job requirements.

Step 1: Input Job Description & Upload Resumes



📝 AI Resume Screening & Candidate Ranking System

Job Description

Enter the job description

Influence and coach a distributed team of engineers.
Facilitate alignment and clarity across teams on goals, outcomes, and timelines.
Manage project priorities, deadlines, and deliverables.
Design, develop, test, deploy, maintain, and enhance large scale software solutions.

📁 Upload Resumes

Upload PDF files

Drag and drop files here
Limit 200MB per file • PDF

test-8.pdf 210.5KB ✕
test-7.pdf 177.7KB ✕
test-6.pdf 155.5KB ✕

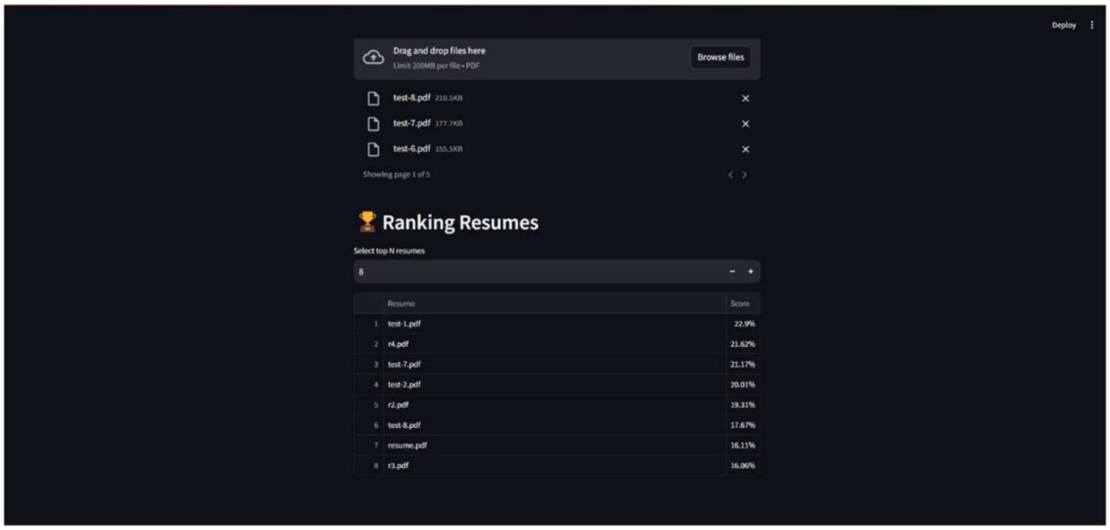
Showing page 1 of 5

Explanation:

- The first step in the system involves entering the job description and uploading candidate resumes. As shown in the snapshot, the interface provides a text box where users can input the job description, detailing the key responsibilities and requirements for the role.

- Below this, there is an upload section that allows users to add multiple resumes in PDF format.
- The system supports drag-and-drop functionality as well as manual file selection, making it convenient for users to upload candidate resumes efficiently

Step 2: Ranking Resume



test-8.pdf 210.5KB
test-7.pdf 177.7KB
test-6.pdf 155.5KB
Showing page 1 of 5

🏆 Ranking Resumes

Select top N resumes: 8

	Resume	Score
1	test-1.pdf	22.9%
2	r4.pdf	21.62%
3	test-7.pdf	21.17%
4	test-2.pdf	20.01%
5	r2.pdf	19.31%
6	test-8.pdf	17.67%
7	resume.pdf	16.11%
8	r3.pdf	16.06%

Explanation:

- The ranking system processes the uploaded resumes and displays the results in a table along with their corresponding scores or percentages.
- Users have the option to select how many top resumes they want to shortlist.
- In the given example, a total of 15 resumes were uploaded, and the system has displayed the top 8 based on their relevance to the job description.

4.2 GitHub Link for Code

Repository Link :
https://github.com/SOHIN-SHAIK/AI-powered-Resume-Screening-and-Ranking-System

Drive Link :
https://drive.google.com/file/d/1LpPHG7h3OV0VZDljGZX4CC0icc4xjtLn/view?usp=drivesdk

CHAPTER 5

Discussion and Conclusion

5.1 Future Work

In the future, the system can be enhanced with additional features to improve its efficiency and accuracy. Some potential improvements include:

- **Advanced NLP and AI Models:** Implementing more sophisticated natural language processing (NLP) techniques and machine learning algorithms to improve resume analysis and ranking.
- **Skill Matching and Weighting:** Allowing users to assign different weights to specific skills and qualifications to refine candidate selection.
- **Integration with Job Portals:** Enabling direct integration with job portals and LinkedIn to fetch candidate profiles dynamically.
- **Automated Feedback System:** Providing AI-generated feedback on resumes to help candidates improve their applications.
- **Multi-format Resume Support:** Supporting additional file formats such as DOCX and TXT for broader compatibility.
- **User Dashboard and Analytics:** Offering a dashboard with analytics on hiring trends, candidate success rates, and system performance.

These enhancements will further optimize the hiring process, making it more efficient and data-driven.

5.2 Conclusion

This project simplifies the hiring process by providing an AI-driven resume ranking system. Instead of manually reviewing each resume, recruiters can upload multiple resumes, input the job description, and let the system rank candidates based on relevance. The ranking is done using automated scoring, ensuring a fair and efficient selection process. The interface is user-friendly, making it easy to shortlist top candidates in just a few clicks. Looking ahead, adding features like AI-driven skill matching and integration with job portals could make it even more powerful. Overall, this system saves time and effort, helping recruiters focus on the best candidates without the hassle of manual screening.

REFERENCES

1. Sridevi G. M, S. Kamala Suganthi : AI based suitability measurement and prediction between job description and job seeker profiles https://doi.org/10.1016/j.jjimei.2022.100109
2. Dr. D. Lakshmi Padmaja, Ch. Vishnuvardhan, G. Rajeev, K. Nitish Sanjeev Kumar : Automated Resume Screening Using Natural Language Processing. https://www.jetir.org/papers/JETIR2303510.pdf
3. Irfan Ali, Nimra Mughal, Zahid Hussain Khand, Javed Ahmed, Ghulam Mujtaba : Resume Classification System using Natural Language Processing and Machine Learning Techniques. https://doi.org/10.22581/muet1982.2201.07