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

Saindla Srinithya, srinithyasridhar@gmail.com

Under the Guidance of

Mr. Soumya Chaudhary

ACKNOWLEDGEMENT

I would like to extend my heartfelt gratitude to my mentor, Mr. Soumya Chaudhary, for his constant support, expert guidance, and invaluable insights throughout this internship. His encouragement and constructive feedback have been instrumental in shaping my approach and deepening my understanding of AI.

This internship has been a remarkable opportunity to apply my knowledge in a real-world setting, enhancing both my technical and problem-solving skills. Working independently on this project has been both challenging and rewarding, providing me with a more profound understanding of AI-driven projects.

I am also grateful to the organization for offering me this opportunity, which has significantly contributed to my growth. This experience has helped me strengthen my academic knowledge while gaining practical expertise in AI applications.

Name: Saindla Srinithya
Email: srinithyasridhar@gmail.com

ABSTRACT

The recruitment process requires significant manual effort in screening resumes, making it both time-consuming and inefficient. To address this challenge, this project presents an AI-powered Resume Screening and Candidate Ranking System designed to streamline the hiring process. By leveraging Natural Language Processing (NLP) and Machine Learning techniques, the system analyzes resumes against job descriptions and ranks candidates based on relevance using cosine similarity. Implemented with Python, Streamlit, and Scikit-learn, it automates resume evaluation, minimizes bias, and enhances hiring efficiency. This report details the system's methodology, implementation, and potential areas for future improvement.

TABLE OF CONTENT

LIST OF FIGURES

CHAPTER 1
Introduction

1.1 Problem Statement:

Manually screening a large volume of resumes poses a significant challenge for recruiters and hiring managers, making the hiring process inefficient, time-consuming, and susceptible to human biases. Traditional candidate shortlisting methods often result in inconsistencies and the potential omission of qualified applicants. Implementing AI-driven automation for resume screening enhances efficiency, ensures accuracy, and promotes fairness in candidate evaluation.

1.2 Motivation:

As the number of job applicants continues to rise, organizations require a fast and reliable method to evaluate resumes without sacrificing quality. AI-driven resume screening minimizes human workload, removes bias, and improves decision-making by ranking candidates according to their relevance to the job description. This project tackles these challenges by utilizing NLP and machine learning techniques.

1.3 Objective:

This project focuses on achieving the following key objectives:

- Developing an AI-powered resume screening system that ranks candidates based on their relevance to the job description.

- Implementing NLP techniques, including TF-IDF vectorization and cosine similarity, for effective textual analysis.

- Designing a user-friendly interface with Streamlit, allowing seamless job description input and resume uploads.

- Enhancing recruitment efficiency by automating the initial screening process.

1.4 Scope of the Project:

To enhance the hiring process, the AI-powered Resume Screening and Candidate Ranking System automates resume evaluation. By leveraging Natural Language Processing (NLP) techniques, it analyzes resumes against job descriptions, ensuring accurate and efficient candidate rankin

In-Scope:

This project adopts a comprehensive and efficient approach to AI-driven resume screening by incorporating the following key aspects:

- **Resume Text Extraction:** Utilizes PDFPlumber to extract textual content from PDF resumes.

- **OCR Support:** Enhances document processing by enabling text extraction from scanned PDFs.

- **NLP-Based Analysis:** Implements TF-IDF vectorization and cosine similarity to analyze job descriptions and resumes.

- **Candidate Ranking:** Ranks resumes based on relevance using similarity measures.

- **Cut-off Feature:** Allows recruiters to set thresholds for selecting top candidates.

- **Automated Processing:** Eliminates manual screening by generating a ranked list of candidates.

- **User Interface & Deployment:** Builds an interactive Streamlit dashboard for recruiters to upload resumes and view rankings.

Out-of-Scope:

The system has the following limitations:
- **Context-Aware Resume Understanding:** It does not incorporate deep learning models or word embeddings like BERT for semantic analysis, relying solely on cosine similarity and TF-IDF.
- **Multilingual Resume Processing**: Currently, it supports only English-language resumes.
- Grammar and Spelling Correction: It does not evaluate grammar quality or correct spelling errors in resumes.
- **Advanced Resume Formatting Analysis:** While it extracts text from PDFs, it does not analyze the structure, design, or layout of resumes.

1. **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) rely heavily on keyword matching, often resulting in inaccurate candidate evaluations due to a lack of contextual understanding. To overcome these limitations, AI-driven techniques 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 interpreting complex sentence structures and contextual relevance. These advancements enable a more precise and accurate ranking process. This project applies NLP techniques, including TF-IDF, cosine similarity, and vectorization, to enhance 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):** Evaluates the significance of words in resumes based on their relevance to the job description, aiding in candidate ranking.

- **Cosine Similarity:** Determines the degree of similarity between resumes and job descriptions by analyzing their vectorized representations.

2. **Word Embedding Models** (Word2Vec):

- Transforms words into vector representations, enabling semantic similarity comparisons instead of relying solely on keyword matching.

3. **Transformer-Based Models** (BERT, RoBERTa, XLNet):

4. **BERT (Bidirectional Encoder Representations from Transformers):** A deep learning model that captures contextual meaning by processing words in both directions.

5. **RoBERTa, XLNet:** Advanced versions of BERT that optimize training methods and improve overall performance.

6. **Large Language Models** (LLMs – GPT, Gemini):

- **LLMs (Large Language Models):** Advanced AI models developed using transformer architectures.
- **GPT and Gemini:** These models effectively analyze resumes by understanding complex sentence structures, relationships, and intent, improving the screening process.

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 depend heavily on keyword matching, often resulting in false positives or negatives due to a lack of contextual understanding.

2. **Limited Context Awareness:** Many AI models struggle to grasp the deeper meaning of resumes and job descriptions, missing subtle relationships.

3. **Bias in Screening**: Pretrained models may inherit biases from their training data, potentially affecting fair candidate evaluation.

4. **Lack of Ranking Mechanisms:** Most existing systems rely on binary filtering (match/no match) rather than ranking candidates based on relevance.

How This Project Addresses These Gaps:

- **Enhanced Candidate Ranking:** The system goes beyond basic filtering by utilizing TF-IDF vectorization and cosine similarity to rank resumes, ensuring a more precise shortlisting process.
- **Minimized Screening Bias:** Automating the evaluation process and standardizing ranking criteria help reduce human biases in candidate selection**.**
- **Application of Advanced NLP Techniques:** Techniques such as TF-IDF, cosine similarity, and vectorization are implemented to enhance the accuracy of 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.
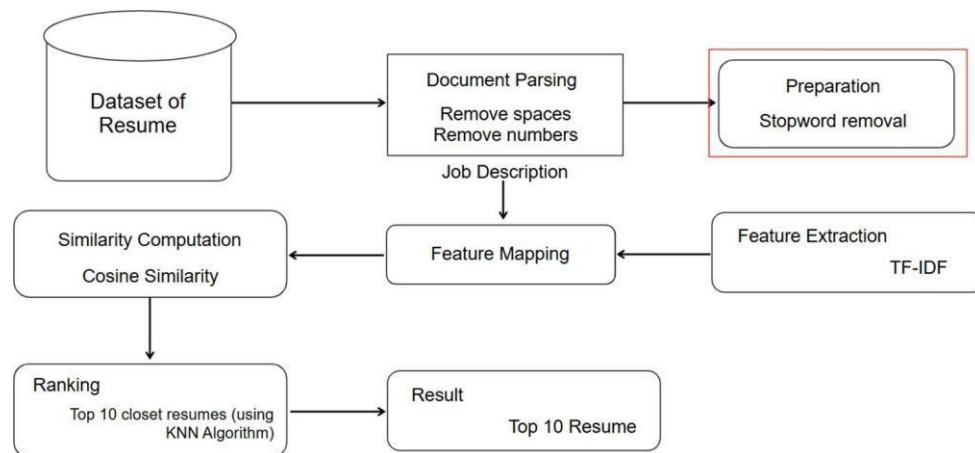
Explanation of the Diagram:

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**

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
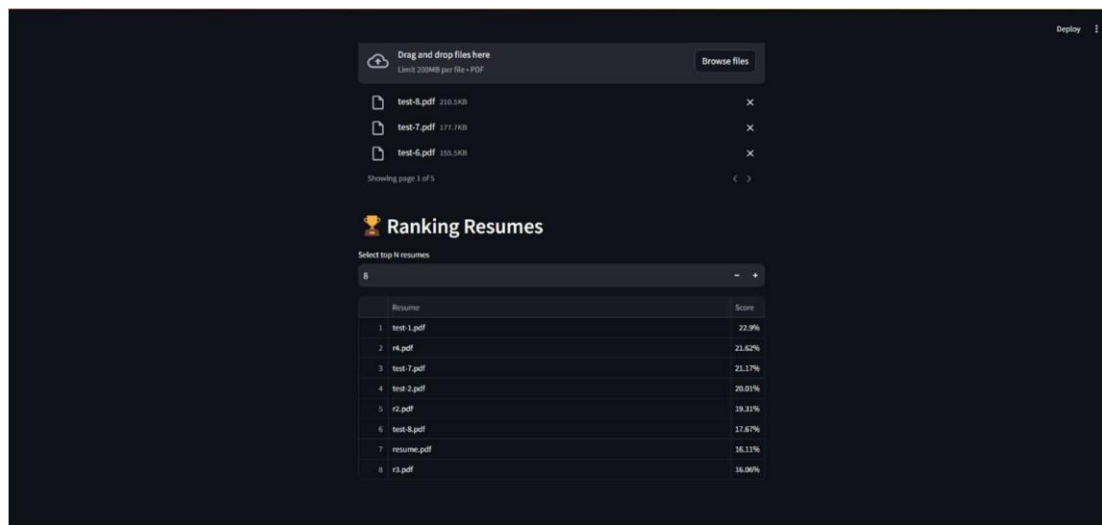
Explanation:

- 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/Srinithya-09/AICTE

CHAPTER 5

Discussion and Conclusion

1.1 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.

1.2 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