Project Title

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

Manish Kujur,

manishkujur05@gmail.com

Under the Guidance of

Saomya Chaudhury

ACKNOWLEDGEMENT

We would like to take this opportunity to express our deep sense of gratitude to all individuals who helped us directly or indirectly during this thesis work.

I would like to express my heartfelt gratitude to Saomya Chaudhury for their invaluable guidance and support throughout my internship project, AI-powered Resume Screening and Ranking System. Their insights and expertise have been instrumental in shaping my understanding of AI and its practical applications.

I am also grateful to my mentors, colleagues, and everyone who provided feedback and encouragement during this journey. Their inputs have helped me refine my approach and improve my learning.

This experience has been a significant step in my growth, and I truly appreciate the opportunity to work on this project.

ABSTRACT

The AI-powered Resume Screening and Ranking System is designed to automate the resume evaluation process by comparing resumes against a given job description. This project leverages Natural Language Processing (NLP) and machine learning techniques to extract, clean, and analyze textual data from resumes in PDF format. By utilizing the TF-IDF (Term Frequency-Inverse Document Frequency) vectorization and calculating cosine similarity, the system efficiently measures the relevance of each resume to the provided job description.

The primary objective is to reduce the time and effort spent on manual resume screening, enabling recruiters to focus on top candidates. The methodology involves text extraction using PyMuPDF (Fitz), text preprocessing using NLTK to remove stopwords and punctuation, and feature vectorization using TfidfVectorizer. The similarity score between the resume and the job description is computed using cosine similarity, providing an objective ranking of candidates.

The application is built using Streamlit for a user-friendly interface, allowing multiple resume uploads and real-time comparison. The resumes are ranked in descending order based on their similarity scores, providing clear insights for recruiters.

In conclusion, this AI-powered system streamlines the hiring process, offering an efficient and accurate solution for resume screening. Future enhancements could include incorporating additional features such as skill extraction, experience classification, and automated shortlisting to further optimize recruitment workflows.

TABLE OF CONTENT

LIST OF FIGURES

CHAPTER 1

Introduction

1.1Problem Statement:

In the modern recruitment landscape, companies often receive hundreds or even thousands of resumes for a single job opening. Manually reviewing and shortlisting candidates is a time-consuming and resource-intensive task, leading to delays in the hiring process. Additionally, human biases and inconsistencies in resume evaluation can result in unfair decision-making and missed opportunities to identify the most suitable candidates.

The problem becomes even more significant for large-scale recruitment drives and organizations with limited HR resources. Relying on traditional resume screening methods not only increases the time-to-hire but also reduces the chances of finding the best talent.

This project aims to address these challenges by developing an AI-powered Resume Screening and Ranking System that automates the resume evaluation process. By using Natural Language Processing (NLP) and Machine Learning (ML) techniques, the system efficiently compares resumes against a given job description and ranks them based on their relevance. This reduces manual effort, minimizes biases, and ensures a faster, more objective hiring process.

The implementation of this AI-based solution provides organizations with a scalable and reliable approach to screening resumes, improving recruitment efficiency and enhancing candidate selection accuracy.

1.2Motivation:

The motivation behind developing the AI-powered Resume Screening and Ranking System stems from the increasing challenges faced by recruiters in managing large volumes of resumes. In a highly competitive job market, companies receive thousands of applications for a single position, making the manual screening process inefficient, time-consuming, and prone to human error.

AI has the potential to transform the hiring process by providing an objective and automated solution for resume evaluation. By leveraging Natural Language Processing (NLP) and Machine Learning (ML), this system can accurately assess candidate qualifications, skills, and experience, ensuring that the most relevant candidates are shortlisted. The reduction in manual effort allows HR professionals to focus on strategic tasks such as interviews and candidate engagement.

Potential Applications:

- Recruitment Agencies: Provides a data-driven approach for efficient talent acquisition.
- ATS Integration: Integrates seamlessly with existing Applicant Tracking Systems (ATS) to provide real-time candidate ranking.

Impact:

- Increased Efficiency: Reduces the time required for resume screening, leading to faster hiring decisions.
- Improved Candidate Experience: Enables quicker feedback to applicants, enhancing the overall recruitment experience.

1.3Objective:

The primary objective of the AI-powered Resume Screening and Ranking System is to develop an automated solution that evaluates and ranks resumes based on their relevance to a given job description using Natural Language Processing (NLP) and Machine Learning (ML) techniques.

The specific objectives of the project include:

- Automated Resume Extraction:
  Extract text from multiple resume PDFs using PyMuPDF (Fitz) for further analysis.
- Text Preprocessing and Cleaning:
  Perform text normalization by removing punctuation, stopwords, and unnecessary characters using NLTK to ensure clean data for analysis.
- Feature Extraction and Representation:
  Convert the cleaned text into numerical representations using TF-IDF (Term Frequency-Inverse Document Frequency) for efficient comparison.
- Similarity Measurement:
  Calculate the similarity between the resumes and the job description using cosine similarity to determine relevance.
- Resume Ranking:
  Rank resumes based on their similarity scores, providing an ordered list of the most relevant candidates.
- User-Friendly Interface:
  Develop a simple and intuitive Streamlit web application for uploading resumes, entering job descriptions, and displaying ranked results.
- Efficient Recruitment Process:

Provide recruiters with a reliable and objective tool to streamline the candidate shortlisting process, reducing manual effort and decision-making time.

1.4Scope of the Project:

The AI-powered Resume Screening and Ranking System is designed to assist recruiters and hiring managers in efficiently evaluating and ranking candidates based on job descriptions. By leveraging Natural Language Processing (NLP) and Machine Learning (ML), the system automates the resume screening process, reducing the time and effort required for manual evaluations.

Scope:

- Resume Text Extraction: The system extracts and processes text from resumes in PDF format using PyMuPDF (Fitz).
- Job Description Analysis: Users can input job descriptions, which will be analyzed and cleaned for comparison.
- Text Preprocessing: The project uses NLTK for text cleaning, including removing stopwords and punctuation, and converting text to lowercase.
- Similarity Calculation: The system applies TF-IDF vectorization and cosine similarity to measure how closely a resume matches a given job description.
- Candidate Ranking: Based on the similarity scores, resumes are ranked in descending order, providing a clear view of the most relevant candidates.
- User Interface: A user-friendly Streamlit web application allows multiple PDF uploads, job description input, and result visualization.

Limitations:

- File Format: Only supports PDF files for resume uploads. Resumes in other formats like DOC or TXT are not currently supported.
- Language Support: The system is designed to process resumes written in English. Multilingual support is not available.

- Limited Context Understanding: The model relies solely on textual data for comparison and does not consider non-textual elements like resume design, images, or certifications in graphical format.
- Job Description Quality: The accuracy of the ranking depends heavily on the clarity and completeness of the job description provided by the user.
- Bias in Data: Since the system learns from text data, any bias present in the training data may influence results. Mitigating this bias is an area for future enhancement.
- No Real-time Feedback: The current version does not offer real-time feedback or suggestions for improving resumes or job descriptions.

Future Enhancements

- Support for multiple file formats such as DOCX and TXT.
- Multilingual resume processing.
- Integration with Applicant Tracking Systems (ATS).
- Incorporation of additional ranking factors like candidate location, experience level, and skill endorsements.

CHAPTER 2

Literature Survey

2.1Review of Relevant Literature

The growing demand for automated hiring solutions has led to extensive research in the field of **AI-powered resume screening**. Several studies and projects have explored the use of **Natural Language Processing (NLP)**, **Machine Learning (ML)**, and **Deep Learning** to enhance the recruitment process. Existing systems often apply **text mining**, **semantic analysis**, and **ranking algorithms** to evaluate candidate resumes against job descriptions.

Key research areas include:

- **Resume Parsing:** Extracting structured information from resumes using text extraction libraries like **PyMuPDF** and **pdfplumber**.
- **Candidate Ranking:** Using similarity metrics such as **cosine similarity** and **Euclidean distance** to rank resumes.
- **NLP Techniques:** Employing methods like **TF-IDF**, **Word2Vec**, and **BERT** to generate feature representations from text.
- **Bias Mitigation:** Developing algorithms to minimize hiring biases and ensure fair recruitment.

2.2Mention any existing models, techniques, or methodologies related to the problem.

Several methodologies have been proposed for automated resume screening:

1. **TF-IDF and Cosine Similarity**
    a. Frequently used in text-based applications to compare resumes and job descriptions.
    b. Efficient for computing text relevance.
    c. **Limitation:** May struggle with understanding contextual meaning.
2. **Machine Learning Classifiers**
    a. Algorithms such as **SVM**, **Naive Bayes**, and **Logistic Regression** are trained on labeled datasets to predict candidate suitability.
    b. **Limitation:** Requires extensive labeled data for training.
3. **Deep Learning Models**

   a. Solutions using models like **BERT** and **GPT** achieve high accuracy in understanding text semantics.
   b. **Limitation:** Computationally expensive and may require GPUs for inference.
4. **Applicant Tracking Systems (ATS)**
   a. Widely used by companies for resume management.
   b. **Limitation:** Often lack sophisticated AI capabilities for accurate ranking.

2.3 Highlight the gaps or limitations in existing solutions and how your project will address them.

Despite advancements in AI-based resume screening, existing solutions face several limitations:

- **Contextual Understanding:** Many models rely on simple keyword matching instead of comprehending the deeper meaning of job descriptions.
- **Bias in Data:** Some algorithms can inadvertently learn biases from historical hiring data.
- **Scalability:** Complex models can be resource-intensive, limiting their applicability for small and medium-sized enterprises.
- **Limited User Control:** Most solutions provide limited customization for recruiters to adjust the ranking criteria.

How This Project Addresses the Gaps

The **AI-powered Resume Screening and Ranking System** proposed in this project overcomes these limitations by:

- **Efficient Text Processing:** Using **TF-IDF vectorization** combined with **cosine similarity** for fast and effective resume comparison.
- **Bias Reduction:** Implementing a transparent and standardized ranking system, reducing subjective biases in candidate evaluation.
- **User-Friendly Interface:** Providing a **Streamlit-based web application** for easy resume upload, job description input, and ranked result display.
- **Scalability:** Designed to handle multiple resume uploads simultaneously without significant resource consumption.

- **Customization:** Allowing recruiters to evaluate similarity scores and interpret results for informed decision-making.

CHAPTER 3

Proposed Methodology

3.1 System Design

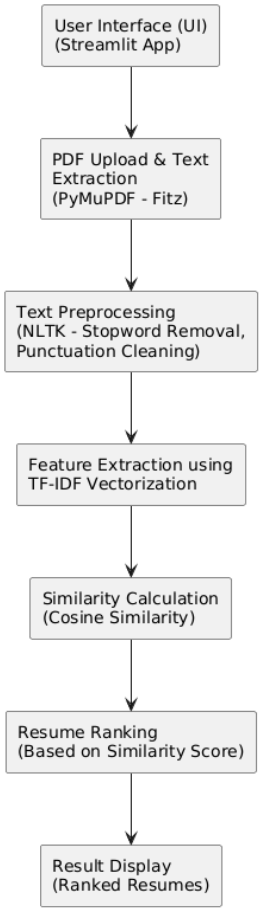


Fig 1: System Design Diagram

Explanation of the Diagram

- User Interface (UI): The user accesses the system using a simple Streamlit-based web interface. They upload multiple resumes in PDF format and input a job description.
- PDF Upload & Text Extraction: The system uses PyMuPDF (Fitz) to extract raw text from the PDF resumes.
- Text Preprocessing: The extracted text is cleaned using NLTK. This involves removing stopwords, punctuation, and unnecessary characters for better analysis.
- Feature Extraction: Using TF-IDF Vectorization, the cleaned text is converted into numerical vectors that represent the importance of words in the resumes and the job description.

- Similarity Calculation: The system computes the cosine similarity between the job description vector and each resume vector to determine relevance.
- Resume Ranking: Based on the similarity scores, resumes are ranked from highest to lowest, representing the most to least relevant candidates.
- Result Display: The ranked resumes are displayed in the Streamlit interface with their corresponding similarity scores.

3.2 Requirement Specification

To implement the proposed solution, the following hardware and software resources are required:

3.2.1 Hardware Requirements:

- **Processor: Intel Core i5 or higher**
- **RAM: Minimum 4 GB (16 GB recommended for larger datasets)**
- **Graphics Card: Not required for this project**
- **Operating System: Windows, Linux, or macOS**

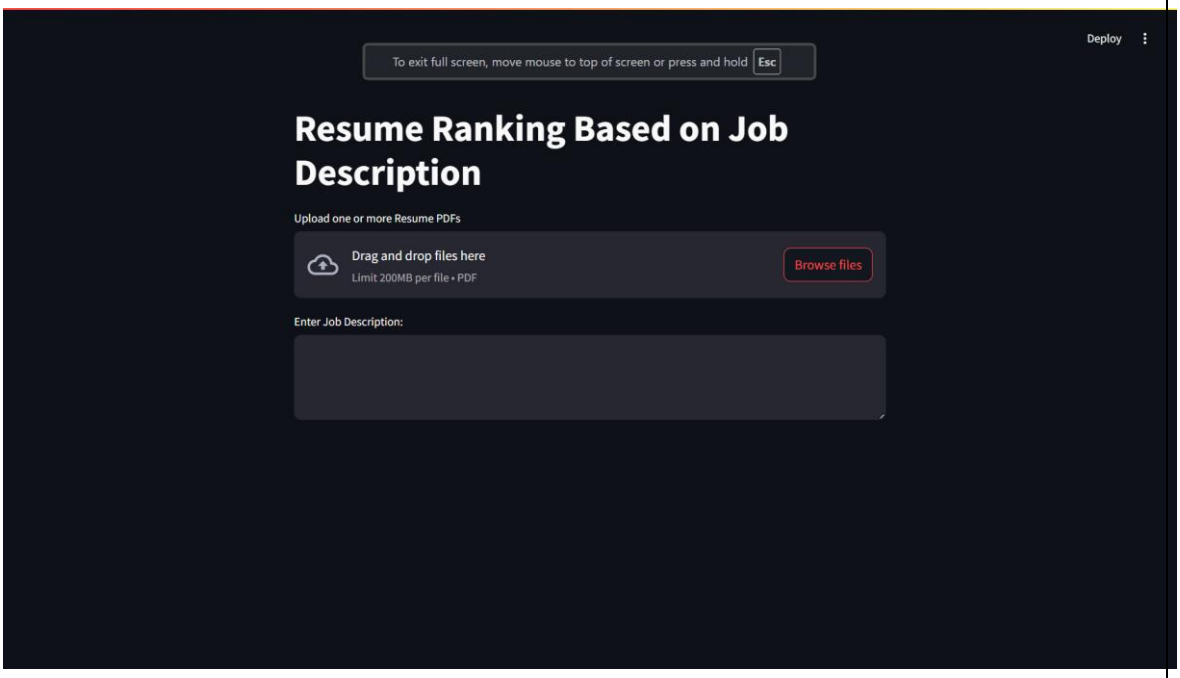
3.2.2 Software Requirements:

- **Programming Language: Python 3.10 or above**
- **Libraries and Frameworks:**
  - **Streamlit (For Web Interface)**
  - **PyMuPDF (Fitz) (For PDF Text Extraction)**
  - **NLTK (For Text Preprocessing)**
  - **NumPy (For Numerical Operations)**
  - **SciKit-Learn (For TF-IDF Vectorization and Cosine Similarity)**
- **Development Tools:**
  - **Visual Studio Code / PyCharm / Jupyter Notebook**
- **Package Manager:**
  - **pip (Python Package Manager)**
- **Additional Tools:**
  - **Git (For Version Control)**

CHAPTER 4

Implementation and Result

4.1 Snap Shots of Result:



- Fig 2: The system is labeled "Resume Ranking Based on Job Description.

  File Upload Section:

  - Users can **upload one or multiple resume PDFs** using the drag-and-drop feature or the **"Browse files"** button.
  - The system supports PDFs with a size limit of **200MB** per file.

Job Description Input Section:

- Users enter the job description in the provided text area.
- The input will be processed using text cleaning and vectorization techniques for comparison.

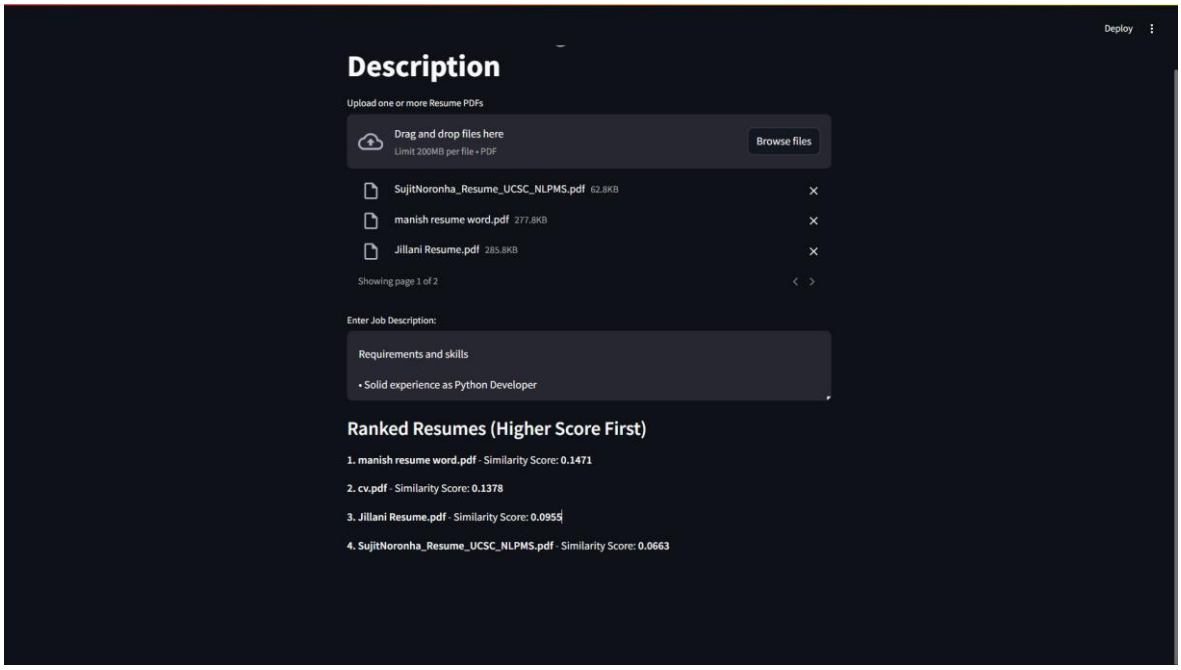


Fig 3: Second Screen (Results Page)

- **File Display:**
  - Uploaded resumes are listed by name along with their file sizes.
  - Users have the option to remove any uploaded files using the "**X**" button.
- **Job Description:**
  - The entered job description is displayed, providing clarity on the input used for similarity calculation.
- **Resume Ranking Results:**
  - After processing, the resumes are **ranked in descending order of similarity** based on the cosine similarity score.
  - The similarity score (e.g., **0.1471**) represents how closely a resume matches the job description. A higher score indicates a better match.
- **File Name and Score:** Each resume is displayed with its corresponding similarity score, ensuring transparency in the evaluation process.

4.2 GitHub Link for Code:

https://github.com/manish3000/Resume-Screening-and-Ranking-System/tree/main

CHAPTER 5

Discussion and Conclusion

5.1 Future Work:

While the current AI-powered Resume Screening and Ranking System efficiently evaluates and ranks resumes based on job descriptions using TF-IDF and cosine similarity, there are several areas for improvement and further development:

1. **Enhanced Context Understanding:**
    a. Implement advanced NLP models like **BERT** or **GPT** to better understand the context and semantics of both resumes and job descriptions.
2. **Multilingual Support:**
    a. Extend the system to support resumes and job descriptions in multiple languages, expanding its usability for global recruitment.
3. **File Format Compatibility:**
    a. Add support for file formats such as **.docx**, **.txt**, and **.rtf** to accommodate resumes in various formats.
4. **Bias Mitigation:**
    a. Implement bias detection algorithms to ensure fairer ranking and minimize the impact of discriminatory patterns in data.
5. **Real-time Feedback:**
    a. Provide users with insights or suggestions on how candidates can improve their resumes to match specific job descriptions more effectively.
6. **Integration with ATS (Applicant Tracking Systems):**
    a. Allow seamless integration with existing ATS platforms to streamline the recruitment workflow.
7. **Advanced Customization:**
    a. Enable recruiters to adjust ranking criteria, such as placing more weight on specific skills or qualifications.
8. **Performance Optimization:**
    a. Implement efficient algorithms to reduce processing time for large-scale recruitment scenarios involving thousands of resumes.

5.2 Conclusion:

The **AI-powered Resume Screening and Ranking System** successfully automates the candidate evaluation process, providing recruiters with a reliable and efficient solution for

shortlisting resumes. By applying **Natural Language Processing (NLP)** techniques like **TF-IDF vectorization** and **cosine similarity**, the system accurately ranks resumes based on their relevance to the given job description.

Key contributions of the project include:

- **Time and Effort Reduction:** The system significantly minimizes manual effort by automating the initial screening process.
- **Objective Evaluation:** It ensures unbiased resume ranking by evaluating candidates solely on the basis of textual relevance.
- **User-Friendly Interface:** The intuitive **Streamlit** application allows recruiters to easily upload resumes, enter job descriptions, and view ranked results.
- **Customizable Solution:** The system provides flexibility for recruiters to make informed decisions by reviewing the similarity scores.

In conclusion, this project demonstrates the potential of AI in transforming the recruitment landscape. With further enhancements, it can become an essential tool for companies aiming to improve hiring efficiency and identify the best talent from a large pool of candidates.

REFERENCES

[1] Minghua Li, Zhongzhi Shi, "From Ontology to Semantic Similarity: Calculation of Ontology-Based Semantic Similarity," *Computational Intelligence and Neuroscience*, vol. 2013, Article ID 793091, 2013.. [https://onlinelibrary.wiley.com/doi/full/10.1155/2013/793091]