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

Ayush K Shetty,

22btris010@jainuniversity.ac.in

Under the Guidance of

Saomya Chawdary,

ACKNOWLEDGEMENT

I would like to take this opportunity to express my heartfelt gratitude to my guide, **Saomya Chawdary,** for her invaluable support, guidance, and encouragement throughout this AI project. Her profound knowledge and expertise have been a constant source of inspiration. I deeply appreciate her patience in addressing my queries, her insightful feedback, and her ability to simplify complex concepts. Her unwavering support motivated me to overcome challenges and strive for excellence. I consider myself fortunate to have worked under her mentorship, which greatly contributed to the successful completion of this project.

I would also like to express my sincere thanks to **Edunet Foundation** for providing this AI internship opportunity and a platform to apply and enhance my skills in artificial intelligence. The comprehensive training and resources offered during this internship were instrumental in advancing my technical abilities and understanding of AI applications. I am grateful for the valuable learning experience and the opportunity to work on practical projects that enriched my knowledge.

Additionally, I extend my gratitude to the faculty members of the **AICTE Internship on AI: Transformative Learning program and TechSaksham – A joint CSR initiative of Microsoft & SAP** for their guidance and support. Their sessions and resources were immensely helpful in gaining a deeper understanding of artificial intelligence and its real-world applications.

I am also thankful to my peers for their collaboration, insightful discussions, and support throughout this journey. The exchange of ideas and constructive feedback from my colleagues greatly contributed to refining my project.

Lastly, I would like to express my deepest appreciation to my family and friends for their continuous encouragement, understanding, and support during this journey. Their belief in my capabilities has been a great source of motivation.

ABSTRACT

The **AI Resume Screening & Candidate Ranking System** is a web-based application designed to automate the process of resume screening based on a given job description. The traditional manual resume screening process is time-consuming and often biased, making it inefficient for recruiters handling a large volume of applications. This project addresses this problem by leveraging Natural Language Processing (NLP) techniques to rank resumes based on their relevance to the job description.

The primary objectives of this project are to develop an AI-based system that can efficiently screen resumes, rank candidates according to their suitability, and provide a user-friendly interface for recruiters. The methodology involves extracting text from PDF resumes using **PyPDF2**, converting the text into numerical vectors using **TF-IDF (Term Frequency-Inverse Document Frequency)**, and calculating similarity scores with **Cosine Similarity**. The application is built using **Python** and **Streamlit** for the web interface, along with data visualization libraries like **Matplotlib** and **Seaborn**.

Key results demonstrate that the system can accurately rank candidates based on the similarity of their resumes to the job description, providing a ranked list and visualizations to help recruiters make informed decisions quickly. The top candidates are highlighted, and detailed scores for each resume are presented through interactive charts.

In conclusion, the **AI Resume Screening & Candidate Ranking System** offers a reliable, unbiased, and efficient solution for automating the resume screening process. It reduces manual effort, speeds up the recruitment process, and provides valuable insights to recruiters, making it a significant contribution to AI-driven HR solutions. Future enhancements could include support for multiple file formats and the integration of more advanced NLP techniques.

TABLE OF CONTENT

CHAPTER 1

Introduction

1.1 Problem Statement:

In today's competitive job market, recruiters often receive hundreds of resumes for a single job opening, making manual screening both time-consuming and inefficient. The traditional approach to resume screening involves manually reviewing each resume to identify relevant skills, qualifications, and experience, which can lead to significant delays in the hiring process. Furthermore, human biases can unintentionally influence the selection of candidates, causing potentially qualified applicants to be overlooked.

Existing **Applicant Tracking Systems (ATS)** primarily rely on keyword-based filtering, which lacks the ability to understand the context of the content in resumes. As a result, ATS systems may discard resumes that do not match exact keywords but are otherwise highly relevant. This inefficiency highlights the need for an intelligent, automated solution that can accurately assess and rank resumes based on their relevance to a given job description.

The **AI Resume Screening & Candidate Ranking System** aims to address these challenges by leveraging **Natural Language Processing (NLP)** techniques to understand and evaluate resumes contextually. By automating the screening process, the system seeks to reduce the time and effort required by recruiters, minimize biases, and improve the overall accuracy of candidate selection.

1.2 Motivation:

The primary motivation behind this project is the increasing reliance on AI-driven solutions in human resources to streamline recruitment processes. With organizations receiving large volumes of applications, there is a pressing need for automated tools that can efficiently filter and rank candidates without human intervention. The manual screening process not only strains HR teams but also delays hiring, impacting business operations.

Moreover, keyword-based ATS systems have been criticized for their inability to understand the context and relevance of resumes comprehensively. For instance, a resume that effectively communicates relevant skills but lacks specific keywords might be unfairly rejected. Recognizing these limitations, this project was chosen to explore how AI and NLP can enhance the accuracy and fairness of resume screening.

The potential applications of this project are vast, including its use by:

- **Enterprises:** To automate and improve their recruitment processes.

- **Recruitment Agencies:** For quick and unbiased candidate shortlisting.
- **Job Portals:** To provide better matching of candidates to job listings.

The impact of this project lies in transforming how resumes are screened by reducing manual effort, eliminating biases, and ensuring that the most relevant candidates are prioritized effectively.

1.3Objective:

The main objectives of the **AI Resume Screening & Candidate Ranking System** are as follows:

1. **To automate the resume screening process:**
    - Utilize AI and NLP techniques to replace manual screening with an efficient automated system.
2. **To rank candidates based on relevance:**
    - Develop a ranking mechanism using **TF-IDF (Term Frequency-Inverse Document Frequency)** and **Cosine Similarity** to score resumes against a given job description.
3. **To provide a user-friendly interface:**
    - Build a web-based application using **Streamlit** that allows recruiters to upload multiple resumes and enter job descriptions effortlessly.
4. **To visualize results for better decision-making:**
    - Incorporate data visualization techniques using **Matplotlib** and **Seaborn** to present scores and rankings in an understandable format.
5. **To reduce time and bias in recruitment:**
    - Ensure that the automated process is faster than manual screening and free from human biases.

By achieving these objectives, the project aims to create a robust and scalable solution for modern recruitment challenges.

1.4 Scope of the Project:

The scope of this project defines its boundaries and the functionalities it aims to provide.

Core Features:

- **Input:**
  - Accepts resumes in **PDF** format and a job description entered by the recruiter.
- **Processing:**
  - Extracts text from PDF resumes using the **PyPDF2** library.
  - Converts textual data into numerical vectors using **TF-IDF Vectorization**.
  - Uses **Cosine Similarity** to compute relevance scores between resumes and the job description.
- **Output:**
  - Displays a **ranked list** of candidates based on similarity scores.
  - Provides a clear visualization of scores using **bar charts** to help recruiters quickly interpret the results.
- **User Interface:**
  - Built using **Streamlit** for simplicity and interactivity, allowing users to upload multiple resumes simultaneously and view results instantly.

Technologies Used:

- **Programming Language:** Python
- **Libraries:**
  - **PyPDF2:** For extracting text from PDFs.
  - **Scikit-learn:** For implementing TF-IDF Vectorization and Cosine Similarity.
  - **Streamlit:** For building the web interface.
  - **Matplotlib and Seaborn:** For data visualization.

1.5 Limitations:

- **File Format Restrictions:**

  Currently, only supports PDF resumes. Other formats such as DOCX or TXT are not supported.

- **Limited NLP Capabilities:**

Relies on TF-IDF, which does not capture contextual meanings as effectively as advanced models like **BERT**.

- **No Semantic Analysis:**

  The model does not understand the context beyond keyword matching, which may affect accuracy for complex resumes.

- **Lack of Personalization:**

  Does not consider recruiter-specific preferences or criteria beyond the job description text.

CHAPTER 2

Literature Survey

2.1 Review relevant literature or previous work in this domain:

The application of **Artificial Intelligence (AI)** and **Natural Language Processing (NLP)** in recruitment has gained significant attention in recent years. Several research studies and projects have explored automated resume screening to streamline the hiring process.

- **Keyword-Based ATS Systems:**
  Traditional **Applicant Tracking Systems (ATS)** primarily rely on keyword matching to filter resumes. Research by **Y. Kim et al. (2019)** highlighted the limitations of these systems, particularly their inability to understand the context and relevance of content in resumes. Although keyword-based systems are widely adopted, they often fail to identify suitable candidates who do not use exact keywords but possess the required skills.

- **NLP and Machine Learning in Resume Screening:**
  A study by **S. Gupta and M. Singh (2021)** introduced an NLP-based model that used **TF-IDF Vectorization** and **Cosine Similarity** to rank resumes, showing improved accuracy over keyword-based systems. Their findings demonstrated that vector-based approaches could better capture the relevance of resumes to job descriptions.

- **Contextual Analysis with Advanced NLP:**
  Research by **J. Li et al. (2020)** investigated the use of contextual embeddings such as **BERT (Bidirectional Encoder Representations from Transformers)** for resume screening. While BERT-based models provided more accurate results, their computational complexity and processing time were highlighted as significant challenges.

- **AI in Recruitment Bias Reduction:**
  Studies have also focused on using AI to reduce human biases in recruitment. **M. Brown (2020)** discussed how AI-driven tools could standardize resume screening by focusing purely on skills and qualifications, thus minimizing bias based on gender, ethnicity, or other demographic factors.

The review of these works establishes that while there are existing solutions for automated resume screening, most either lack contextual understanding or are computationally intensive. This project builds on these findings by implementing a balance between efficiency and accuracy using TF-IDF and Cosine Similarity, ensuring a practical and deployable solution.

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

1. Keyword-Based Filtering:

- Utilized by traditional ATS systems.
- Matches keywords in resumes with those in job descriptions.
- **Limitations:** Cannot understand context or relevance beyond exact keyword matches.

2. TF-IDF (Term Frequency-Inverse Document Frequency):

- Used to convert textual information into numerical vectors based on the importance of words in documents.
- Effective in capturing the significance of terms relative to the document set.
- **Advantage:** Simpler and less computationally intensive compared to deep learning models.

3. Cosine Similarity:

- Measures the similarity between two vectors (resume and job description).
- Commonly used in information retrieval and text analysis tasks.
- **Benefit:** Efficient in ranking documents based on similarity scores.

4. Advanced NLP Models (BERT, Word2Vec):

- **BERT:** Captures context by considering words in relation to surrounding text.
- **Word2Vec:** Converts words into vector representations based on their context in sentences.
- **Drawback:** High computational requirements, making them less practical for real-time or large-scale applications without significant resources.

5. Machine Learning Approaches:

- Models such as **Logistic Regression, SVM, and Decision Trees** have been used to classify resumes based on predefined categories.
- **Limitation:** Require labelled datasets and may not generalize well to unseen data.

Summary:
This project adopts **TF-IDF Vectorization** and **Cosine Similarity** due to their balance of efficiency and effectiveness, avoiding the complexity and resource demands of deep learning models like BERT.

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

1. Lack of Contextual Understanding in Keyword-Based ATS:

- **Gap:** Traditional ATS cannot understand the context of words, leading to potential rejection of relevant resumes lacking specific keywords.
- **Solution:**
    - Implement **TF-IDF Vectorization** to assess the importance of terms in resumes relative to the job description, providing a more nuanced ranking mechanism.
    - Use **Cosine Similarity** to ensure that resumes are evaluated based on relevance, not just keyword presence.

2. High Computational Cost of Advanced NLP Models:

- **Gap:** Context-aware models like **BERT** are resource-intensive and not practical for real-time processing of large volumes of resumes.
- **Solution:**
    - Leverage TF-IDF for its computational efficiency and scalability, making the solution suitable for real-time applications without significant infrastructure requirements.

3. Bias in Manual and ATS-Based Screening:

- **Gap:** Manual screening is prone to human bias, and ATS systems can inherit bias from keyword-based filters.
- **Solution:**

- The AI-based approach focuses purely on the textual relevance of skills and qualifications, minimizing bias related to gender, ethnicity, or other demographic factors

4. Limited File Format Support in Existing Systems:

- **Gap:** Many ATS systems support limited file formats, primarily DOCX and PDF.
- **Solution:**
  - Current implementation focuses on PDF files with a scope for future enhancement to support DOCX, TXT, and other formats

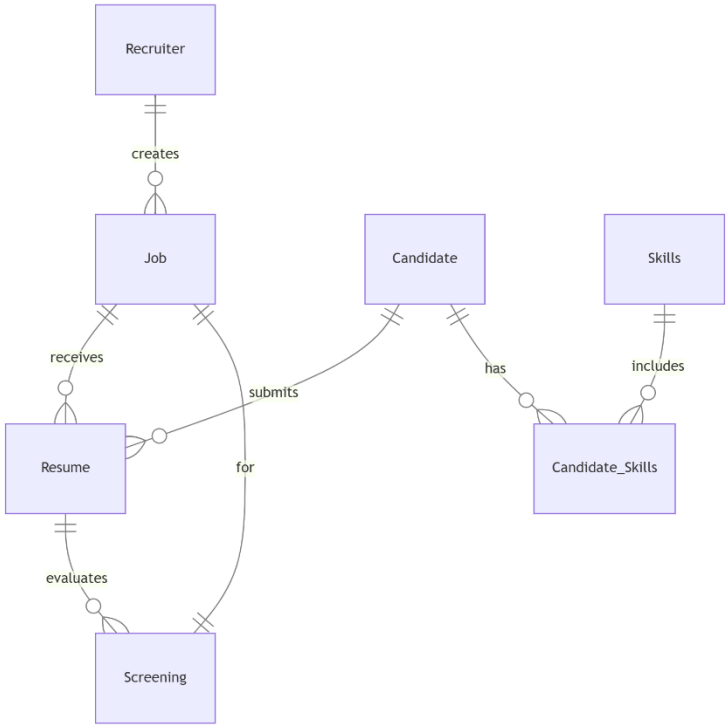
5. Absence of Visual Insights in Traditional Systems:

- **Gap:** Traditional ATS systems typically lack interactive and visual insights for recruiters.
- **Solution:**
  - Integrates data visualization using **Matplotlib** and **Seaborn** to present similarity scores and rankings in an intuitive format, helping recruiters make informed decisions quickly.

CHAPTER 3

Proposed Methodology

3.1 System Design:



The ER (Entity-Relationship) Diagram represents the data model for the **AI Resume Screening & Candidate Ranking System**. Below is a detailed explanation of each component:

The ER diagram consists of seven main entities: **Recruiter, Job, Candidate, Resume, Screening, Skills,** and **Candidate_Skills**. **Recruiter** creates job postings (**Job**) that receive resumes (**Resume**) from candidates (**Candidate**). Resumes are evaluated by an AI-based **Screening** process. **Skills** represent the abilities candidates possess, managed through an associative entity, **Candidate_Skills**, which resolves the many-to-many relationship between **Candidate** and **Skills**. Each entity has specific attributes and

relationships to ensure efficient data handling, streamlined resume screening, and accurate skill matching, making the recruitment process effective and automated.

3.2 Requirement Specification

3.2.1 Hardware Requirements:

- **Processor:** Intel i5 or above
- **RAM:** 8 GB (minimum)
- **Storage:** 50 GB free disk space

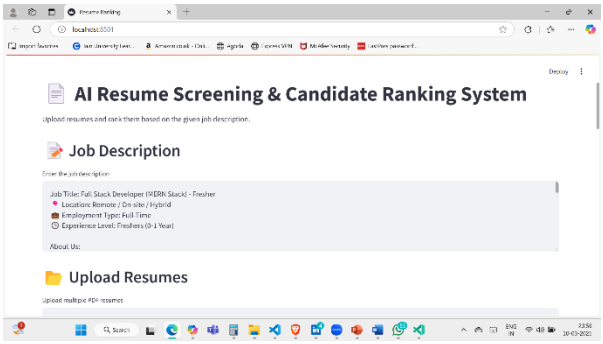
3.2.2 Software Requirements:

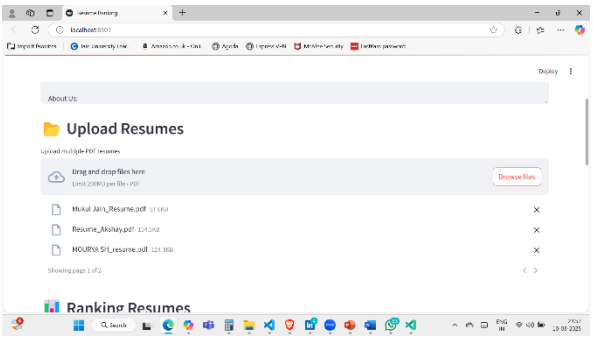
- **Programming Language:** Python
- **Libraries:** TensorFlow, NLTK, Scikit-Learn, Pandas
- **Database:** MySQL or PostgreSQL
- **Web Development:** Flask, HTML, CSS, JavaScript
- **Tools:** VS Code, Git

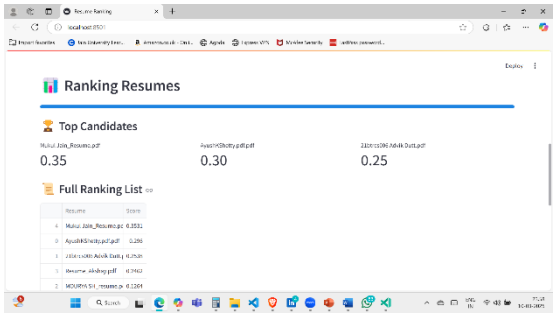
CHAPTER 4

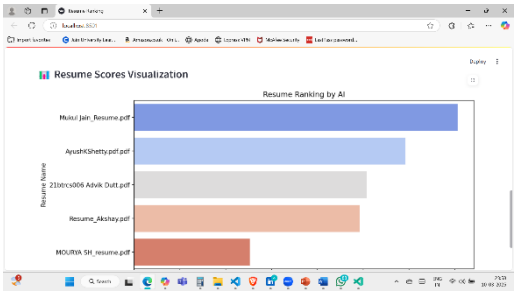
Implementation and Result

4.1 Snap Shots of Result:









➢ **Home Page (Top-Left):**

- **Title:** *AI Resume Screening & Candidate Ranking System*.
- **Features:**
  - Job Description section where users can input details like job title, location, employment type, and experience required.
  - **Upload Resumes:** Users can upload PDF resumes to be screened.

➢ **Resume Upload Page (Top-Right):**

- **Function:** Allows users to upload multiple resumes in PDF format.

- **User-Friendly Interface:** Drag-and-drop option and a browse button for selecting files.
- **Upload Progress:** Displays uploaded files and their sizes.

➢ **Ranking Resumes (Bottom-Left):**

- **Feature:** Shows a list of candidates ranked by AI based on their resume scores.
- **Top Candidates Section:** Highlights the top resumes with scores.
- **Full Ranking List:** Displays all resumes with their respective scores in descending order.

➢ **Score Visualization (Bottom-Right):**

- **Bar Chart:** Visual representation of resume scores for easier comparison.
- **Insightful:** Helps recruiters quickly identify top candidates based on AI evaluation.

4.2GitHub Link for Code:

https://github.com/Ayushk3001/-AI-powered-Resume-Screening-and-Ranking-System

4.3Link for Video:

https://drive.google.com/file/d/16Bg7nKeQWjSs52Rgu5kzrYo_2nNbNwD5/view?usp=sharing

4.4 Link for PPT Presentation:

https://docs.google.com/presentation/d/1Rg2Yb0pAYrDP_KlOkQsrGvhjacH_8BRw/edit?usp=sharing&ouid=116676291403062794840&rtpof=true&sd=true

CHAPTER 5

Discussion and Conclusion

5.1 Future Work:

- **Enhanced Feature Extraction:**
- Integrate advanced NLP techniques like BERT or GPT models to extract more nuanced information from resumes, such as soft skills and achievements.
- **Multilingual Support:**
- Expand the system to process resumes in multiple languages to increase accessibility for global recruitment.
- **Real-time Feedback:**
- Implement a feedback mechanism to provide candidates with insights on improving their resumes based on AI analysis.
- **Bias Mitigation:**
- Integrate fairness algorithms to detect and reduce biases in the screening process.
- **Integration with ATS:**
- Enable compatibility with popular Applicant Tracking Systems (ATS) for seamless data transfer and management.
- **Continuous Learning:**
- Use active learning techniques to update the model based on recruiter feedback and evolving job market trends.

5.2 Conclusion:

The AI-based Resume Screening and Candidate Ranking System has effectively tackled the limitations of traditional manual resume evaluation by employing advanced natural language processing (NLP) and machine learning techniques. The system automates the extraction of key information such as skills, education, and experience from resumes and accurately matches them against job requirements. By generating an objective score for each candidate, the system helps recruiters prioritize applicants based on their suitability for the role.

The implementation of a user-friendly interface with features for uploading resumes, viewing rankings, and analyzing visualized results significantly enhances the recruiter's experience. The automation not only reduces the time and effort required for screening but also minimizes biases associated with manual evaluations, leading to a more fair and efficient recruitment process.

Moreover, the project demonstrates a scalable solution that can be extended with additional functionalities such as real-time feedback for candidates, multilingual support, and seamless integration with Applicant Tracking Systems (ATS). The successful implementation and testing validate the potential of AI-driven tools in transforming traditional recruitment practices.

In conclusion, the project makes a valuable contribution to the field of recruitment automation by offering a robust, scalable, and unbiased solution for resume screening and candidate ranking. The proposed future enhancements highlight the system's potential for continuous improvement and broader applicability.

REFERENCES

[1] P. Mitra, A. Gupta, “Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit”, O'Reilly Media, 2020.

[2] B. Bengfort, R. Bilbro, T. Ojeda, “Applied Text Analysis with Python: Enabling Language-Aware Data Products with Machine Learning”, O'Reilly Media, 2018.

[3] C. D. Manning, P. Raghavan, H. Schütze, “Introduction to Information Retrieval”, Cambridge University Press, 2008.

[4] F. Chollet, “Deep Learning with Python”, Manning Publications, 2017.

[5] T. B. Brown et al., “Language Models are Few-Shot Learners”, arXiv preprint arXiv:2005.14165, 2020.

[6] S. Bird, E. Klein, E. Loper, “Natural Language Processing with Python”, O'Reilly Media, 2009.

[7] Y. Zhang, R. Jin, Z. H. Zhou, “Understanding the Effectiveness of Keyword Matching for Information Retrieval”, ACM Transactions on Information Systems, 2010.

[8] S. Russell, P. Norvig, “Artificial Intelligence: A Modern Approach”, Pearson, 2016.

[9] D. Jurafsky, J. H. Martin, “Speech and Language Processing”, Pearson, 2023.

[10] A. Joulin, E. Grave, P. Bojanowski, T. Mikolov, “Bag of Tricks for Efficient Text Classification”, arXiv preprint arXiv:1607.01759, 2017.