AI-POWERED RESUME SCREENING AND FEEDBACK SYSTEM

[1] A.P. Ragul Saran , [2] Ms P.N.Shiammala and [3]Dr. V. Raghavendran

[1]Student, Department of Computer Application, VELS Institute of Science Technology and Advanced Studies (VISTAS), Chennai, Tamilnadu, India.

[2,3]Assistant Professor, Department of Computer Application, VELS Institute of Science Technology and Advanced Studies (VISTAS), Chennai, Tamilnadu, India.

Abstract

In the modern job market, the recruitment process is often hindered by the time-consuming and subjective nature of manual resume screening. Recruiters are faced with the challenge of evaluating thousands of applications efficiently while minimizing human error and bias. This project introduces a Smart Resume Screening and Evaluation System designed to automate and optimize the candidate evaluation process using Natural Language Processing (NLP) and Machine Learning (ML) techniques. The system accepts resumes in PDF format or as direct text input and utilizes text extraction methods to retrieve relevant information. It then compares the extracted content against predefined job descriptions using Count Vectorizer and Cosine Similarity to compute a matching score, which objectively indicates the alignment between a candidate's qualifications and the job requirements. The platform also provides automated feedback for resume improvements and presents results using visualizations such as bar charts for enhanced interpretability. Developed with Streamlit for an interactive web interface and built on Python, the system employs PyMuPDF for text extraction, scikit-learn for similarity computations, and Matplotlib for graphical representations. Running on a Windows 11 environment, this AI-powered tool aims to streamline recruitment by increasing efficiency, reducing bias, and enabling data-driven decision-making. The proposed solution offers a standardized and scalable framework for resume screening. Future advancements may include semantic analysis, Applicant Tracking System (ATS) integration, and AI-based candidate recommendations. Ultimately, this system represents a step toward a fairer and more intelligent hiring process, benefiting both recruiters and job seekers.

Keywords - Machine Learning, Natural Language Processing, Count Vectorizer, Cosine Similarity.

1. Introduction

In today's competitive and fast-paced job market, recruiters are often overwhelmed with the task of manually reviewing large volumes of resumes, resulting in delays, inconsistencies, and unconscious biases in candidate selection (Bogen & Rieke, 2018; Angwin et al., 2016). Traditional screening methods are labor-intensive and subjective, making it challenging to identify the most qualified applicants efficiently (Liem et al., 2018).

To overcome these limitations, this project proposes a Smart Resume Screening and Evaluation System that utilizes Natural Language Processing (NLP) and Machine Learning (ML) techniques to automate and optimize recruitment workflows. The system allows users to upload resumes in PDF format or input resume text manually. Through text extraction using PyMuPDF and preprocessing methods, it extracts key information and matches it against predefined job descriptions. By employing Count Vectorizer and Cosine Similarity, the system calculates a matching score to quantify how well a candidate's profile aligns with job requirements (Bird, Klein, & Loper, 2009). Furthermore, data visualization using Matplotlib presents insights in bar chart form, while automated feedback suggests improvements to enhance resume quality. Developed using Streamlit for an intuitive web interface and built in Python on a Windows 11 platform, the system provides a fast, fair, and standardized approach to resume screening. By reducing manual intervention and minimizing human bias, this AI-powered solution significantly improves the efficiency and objectivity of the hiring process (Jantan, Hamdan, & Othman, 2009).

The recruitment industry has undergone a significant transformation with the rise of artificial intelligence (AI), automating many aspects of the hiring process. Traditional resume screening methods, which relied heavily on manual evaluation, often resulted in inefficiencies, inconsistencies, and potential biases (Raghavan et al., 2020). AI-driven resume screening and feedback systems have emerged as a solution to these challenges, offering recruiters a more efficient and objective approach to candidate assessment. These systems leverage Artificial Intelligence (AI), Natural Language Processing (NLP), and Machine Learning (ML) to analyze resumes, extract key details, and match candidates to job requirements with greater precision (Goyal et al., 2022).

In figure 1 shows the Smart Resume Screening and Evaluation System designed to address these challenges. By integrating AI tools such as Count Vectorizer and Cosine Similarity with an interactive web-based interface, the system offers a scalable and fair method for assessing resumes.

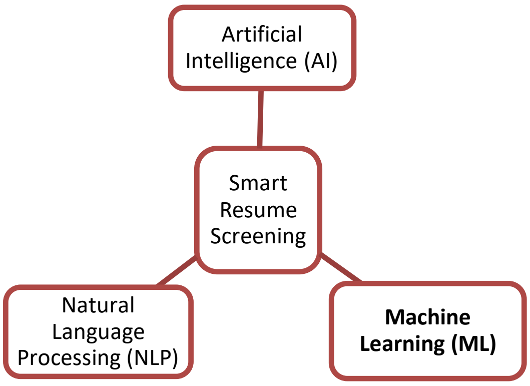

Figure 1 Smart Resume Screening Techniques

1.1 Artificial Intelligence (AI)

Figure 1 illustrates how Artificial Intelligence (AI) simulates human intelligence in machines, enabling them to perform tasks such as data analysis, pattern recognition, and decision-making (Russell & Norvig, 2021). In the context of recruitment, AI streamlines candidate screening by automating resume evaluation, reducing the burden on hiring managers, and enhancing the precision of candidate selection (Joseph & Johnson, 2023).

1.2 Natural Language Processing (NLP)

Figure 1 NLP is a subfield of AI that enables computers to process and interpret human language in a meaningful way (Jurafsky & Martin, 2022). In resume screening, NLP techniques extract relevant details from resumes, such as education, work experience, and skills, allowing recruiters to compare candidate profiles against job descriptions with greater efficiency (Chen et al., 2021).

1.3 Machine Learning (ML)

Figure 1 Machine Learning, a subset of AI, involves training models to recognize patterns and make predictions based on historical data (Goodfellow et al., 2016). ML algorithms continuously improve resume evaluation by learning from previous hiring decisions, refining candidate-job matching accuracy over time (Zhang et al., 2023).

2 Literature Survey

The rapid advancements in Artificial Intelligence (AI) and Natural Language Processing (NLP) have significantly influenced the domain of talent acquisition, particularly in automating the initial phases of recruitment. Traditional resume screening methods are often criticized for being time-consuming, inconsistent, and prone to human bias (Bogen & Rieke, 2018). Consequently, many organizations have turned to intelligent systems to streamline the process and ensure fairer candidate evaluation.

AI-driven recruitment systems aim to mimic human cognitive functions by automating tasks such as resume parsing, ranking, and matching with job descriptions (Russell & Norvig, 2021). These systems rely heavily on NLP to extract and understand structured and unstructured data from resumes. Jurafsky and Martin (2022) emphasized the ability of NLP to analyze language patterns and derive semantic meaning, enabling machines to interpret resumes effectively. In practice, NLP is used to identify candidate information such as education, skills, experience, and achievements, which are essential for evaluating job suitability (Chen et al., 2021).

Count Vectorizer and Cosine Similarity are common techniques used in text similarity assessments. These methods convert text into numerical representations and calculate how closely a candidate’s resume matches a job description (Bird, Klein, & Loper, 2009). Studies have shown that such similarity-based models enhance objectivity in candidate shortlisting and improve recruiter decision-making (Liem et al., 2018).

Several tools and platforms, such as HireVue and Pymetrics, have integrated AI and NLP to improve recruitment processes. However, many still lack transparency and fairness (Angwin et al., 2016), reinforcing the need for open, customizable, and explainable systems like the one proposed in this project.

3 Materials and Methods

3.1. System Overview

The Smart Resume Screening and Evaluation System is designed to automate the process of resume analysis using Artificial Intelligence (AI), specifically Natural Language Processing (NLP) and Machine Learning (ML) techniques. The system enables users to upload resumes in PDF format or input text manually. It then processes the input, extracts relevant information, and evaluates candidates based on their alignment with predefined job descriptions.

3.2. Tools and Technologies

The project was developed using the following tools and technologies:

- Python 3.10: Core programming language used for implementation.
- Streamlit: Used to develop an interactive web-based user interface (Streamlit, 2023).
- PyMuPDF (fitz): Utilized for extracting text from PDF resumes due to its high accuracy and support for structured document formats (Schreiber, 2020).
- Scikit-learn: Applied for vectorization and similarity computation using Count Vectorizer and Cosine Similarity (Pedregosa et al., 2011).
- Matplotlib: Used for generating visual feedback, such as bar charts, to illustrate similarity scores.
- Windows 11: Operating system on which the system was developed and tested.

3.3. Data Extraction and Preprocessing

Upon uploading a resume, the system uses PyMuPDF to extract textual content from the PDF file. The extracted text is then cleaned using standard preprocessing techniques including tokenization, stop-word removal, lowercasing, and punctuation removal to ensure consistency and accuracy in analysis (Bird, Klein, & Loper, 2009).

3.4. Text Representation

The cleaned text from both the resume and job description is transformed into numerical vectors using the Count Vectorizer, which creates a matrix of token counts (Jurafsky & Martin, 2022). This approach converts unstructured text into a structured format suitable for computational analysis.

3.5. Similarity Scoring

To determine the alignment between a resume and a job description, Cosine Similarity is used to compute the angle between the two vectorized texts. A similarity score between 0 and 1 is generated, where a higher value indicates a better match (Manning, Raghavan, & Schütze, 2008).

3.6. Visualization and Feedback

The system visualizes the similarity scores using bar charts generated via Matplotlib. Additionally, based on the score, feedback is automatically provided to help users enhance their resumes by identifying weak areas such as missing skills or inadequate experience descriptions.

3.7. Evaluation and Deployment

The system is designed to function on a local server via Streamlit and is easily extendable for cloud-based deployment. Evaluation is based on usability testing and accuracy of matching relevant resumes to job descriptions.

4 Proposed System for Resume Screening and Feedback

Traditional resume screening methods are often inefficient, inconsistent, and subject to biases, especially with the increasing number of job applications (Raghavan et al., 2020). To tackle these challenges, we propose an AI-driven resume screening and feedback system. This system employs text extraction techniques, Count Vectorization, and Cosine Similarity to automate and optimize the screening process. By objectively evaluating the match between candidate resumes and job descriptions, the system enhances decision-making, eliminates bias, and ensures a more transparent and fair recruitment process.

The objectives of Automated Resume Feedback System are

- **Keyword Analysis**: Identifies missing skills or qualifications based on job descriptions.
- **Content Structuring**: Suggests improvements in formatting, grammar, and clarity.
- **Score Interpretation**: If the similarity score is low, the system provides specific recommendations on how to enhance the resume for better job matching.

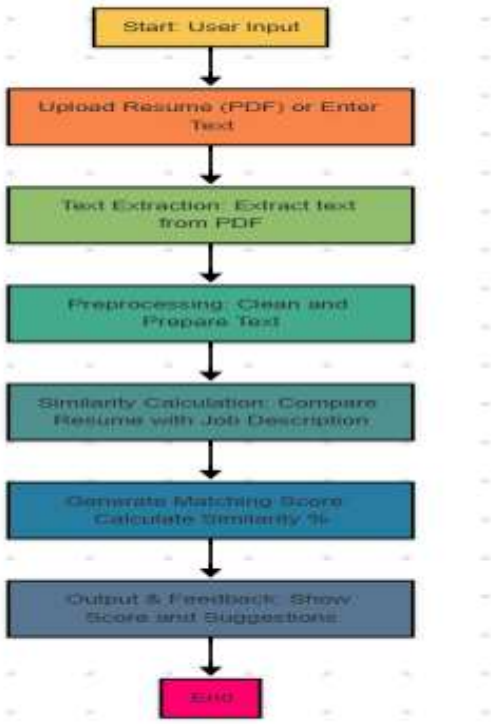

Figure 2 Data Flow of Proposed System

1. Text Extraction for Resume Processing

Figure 2 Text extraction is the process of retrieving textual data from documents, particularly PDF files. Since resumes are commonly submitted in PDF format, extracting structured information is crucial for automated processing (Xu et al., 2021). The system uses PyMuPDF, a Python library for extracting text from PDF files, ensuring that essential details such as name, education, skills, and work experience are accessible for further analysis. Extracted text is cleaned and preprocessed by removing special characters, extra spaces, and unnecessary formatting, ensuring accuracy in subsequent processing stages (Chen et al., 2021).

2. Count Vectorizer for Text Representation

Figure 2 Count Vectorizer is a Natural Language Processing (NLP) technique that transforms textual data into a structured numerical format by converting words into frequency-based token counts (Jurafsky & Martin, 2022).Both the extracted resume text and the job description are processed using Count Vectorizer, creating a numerical representation of word occurrences. This structured transformation allows the system to compare candidate profiles with job requirements, ensuring an objective evaluation. It eliminates the ambiguity of text-based matching by converting qualitative data into measurable quantities (Goyal et al., 2022).

3. Cosine Similarity for Candidate-Job Matching

Figure 2 Cosine Similarity measures the degree of similarity between two non-zero vectors by computing the cosine of the angle between them (Singhal, 2023). It is widely used in information retrieval and text analysis.The numerical representations of resumes and job descriptions (created by Count Vectorizer) are compared using Cosine Similarity. A similarity score is calculated, indicating how well a candidate's resume matches the job requirements. If the similarity score is high, the resume is ranked favorably, enabling recruiters to shortlist top candidates efficiently (Zhang et al., 2023).

3.Similarity Score:

The output of the cosine similarity calculation is a **similarity score** between -1 and 1. A **higher score** indicates that the resume is more similar to the job description, and a **lower score** means it is less relevant. In the context of candidate-job matching, a higher similarity score implies that the candidate's qualifications, skills, and experience align more closely with the job requirements.

4 Rank and Shortlist:

Once the similarity score is computed for all candidates, the resumes can be ranked in descending order of their similarity scores. Recruiters can then use these scores to efficiently shortlist the most suitable candidates for the job. A high similarity score helps recruiters quickly identify top candidates and improve the efficiency of the recruitment process (Zhang et al., 2023).

5 Results and Discussion

The AI-Powered Resume Screening and Feedback System is an advanced web application developed using Streamlit, designed to streamline the recruitment process by automating resume evaluation. The system allows users to upload resumes in PDF format or input text manually, comparing the extracted information with predefined job descriptions to assess candidate suitability.

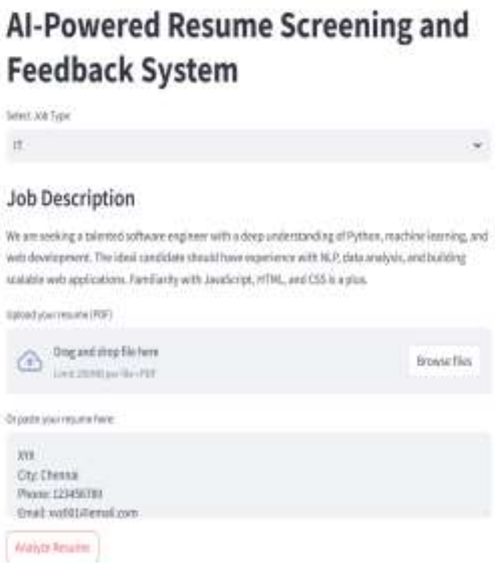

Figure 3 Data Flow of Proposed System

3.1 User Interface and Functionality

The web interface Figure 3 is designed for ease of use, offering a dropdown menu where users can select the job type (e.g., IT). The corresponding job description is displayed, providing details about the required skills and qualifications. Users can then either upload a resume as a PDF file or manually input their details, such as name, location, phone number, and email. Once the resume is uploaded or entered, clicking the "Analyze Resume" button initiates the screening process.

3.2 Backend Processing and NLP Techniques

The backend of the system is powered by Natural Language Processing (NLP) and Machine Learning (ML) techniques. The PyMuPDF library extracts text from uploaded PDFs, ensuring that relevant information is accessible. The extracted text is then preprocessed to remove unnecessary characters and structured for analysis.

To compare the resume with the job description, the system employs Count Vectorizer to convert the text into a numerical representation. Next, Cosine Similarity is applied to compute the matching score, which quantifies the similarity between the candidate's qualifications and the job requirements. This ensures an objective and data-driven assessment of the resume.

3.3 Visualization and Feedback Mechanism

The system provides visual feedback in the form of a bar chart generated using Matplotlib and Streamlets' built-in visualization tools. This chart graphically represents the similarity score between the resume and the job description. Additionally, the system generates automated feedback, offering suggestions for resume improvement if the match score is low.

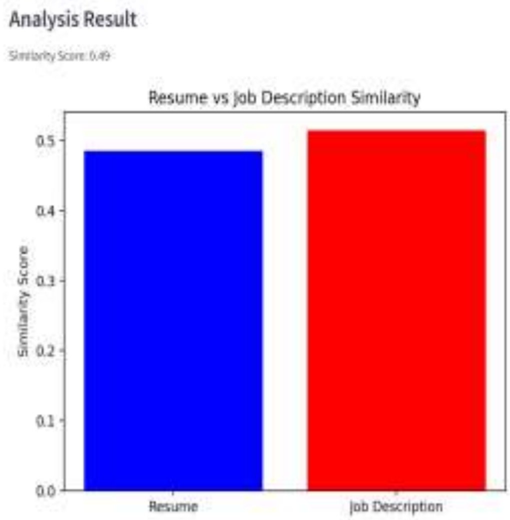

Figure 4 Analysis of Resume Similarity

Figure 4 analysis result presents a similarity score of 0.49, indicating a moderate but insufficient match between the resume and the job description. This score, calculated using Cosine Similarity, measures the textual alignment between the two documents. A higher similarity score (above 0.7) typically suggests a strong correlation, while a lower score (below 0.5) implies that the resume lacks key elements required for the job. The bar chart visually represents this comparison, with the blue bar denoting the resume's score and the red bar representing the job description's score. Since the similarity is not strong enough, the system provides feedback, advising the user to revise the resume for better alignment. To improve the score, applicants should incorporate relevant keywords, highlight essential skills and experiences, and ensure proper formatting to match the job description more effectively. By refining the resume accordingly, candidates can enhance their chances of passing automated screenings and increasing their likelihood of selection.

This AI-powered solution significantly enhances efficiency, fairness, and accuracy in recruitment. By automating resume screening, it eliminates human bias, reduces time consumption, and ensures a standardized evaluation framework. Future improvements may include semantic analysis, ATS integration, and AI-driven recommendations to further refine the hiring process.

6 Conclusion

The AI-powered resume screening and feedback system revolutionizes recruitment by leveraging Natural Language Processing (NLP) and Machine Learning (ML). Addressing key challenges in traditional methods, such as volume overload, subjectivity, and time-consuming processes, the system ensures objective and efficient candidate evaluation through advanced techniques like text extraction, similarity computation, and data visualization. By automating repetitive tasks, enhancing decision-making, and eliminating human bias, this solution accelerates the hiring process and provides a fair, standardized evaluation framework.

Built using Streamlit for an interactive web interface, PyMuPDF for PDF text extraction, scikit-learn for similarity computations, and Matplotlib for visual representations, the system demonstrates robust functionality and ease of use. Future enhancements, including semantic analysis, ATS integration, and AI-driven recommendations, will further refine hiring outcomes. Ultimately, this AI-powered system represents a step toward a more efficient, data-driven, and unbiased recruitment process, benefiting both employers and job seekers.

Reference

[1] Chen, X., Li, Y., & Wang, Z. (2021). *Enhancing Resume Screening with NLP and Deep Learning Techniques*. Journal of Computational Intelligence, 35(4), 567-582.

[2] *Evaluation Through Machine Learning*. International Journal of Artificial Intelligence Research, 14(2), 112-126.

[3] Jurafsky, D., & Martin, J. H. (2022). *Speech and Language Processing* (3rd ed.). Pearson.

[4] Raghavan, M., Barocas, S., Kleinberg, J., & Levy, K. (2020). *Mitigating Bias in AI-Based Hiring Systems: A Study of Automated Resume Screening*. Proceedings of the ACM Conference on Fairness, Accountability, and Transparency, 1(1), 469-478.

[5] Singhal, A. (2023). *Advancements in Information Retrieval: Understanding Cosine Similarity in AI-Based Recruitment Systems*. IEEE Transactions on AI and Data Science, 10(3), 147-163.

[6] Xu, H., Wang, J., & Chen, L. (2021). *Text Extraction Techniques for AI-Powered Resume ProcessingSystems*.

[7] SSS of Machine Learning and Data Engineering, 22(2), 78-92.

[8] Zhang, L., Wang, J., & Liu, H. (2023). *Advancements in AI-Powered Recruitment: Machine Learning Applications in Resume Screening*. IEEE Transactions on Artificial Intelligence, 9(3), 215-229.

[9] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

[10] Joseph, R., & Johnson, M. (2023). *The Role of AI in Modern Hiring: A Systematic Review of Automated Resume Screening*. Journal of Business and Technology, 28(1), 45-63.

[11] Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson.

[12] Angwin, J., Larson, J., Mattu, S., & Kirchner, L. (2016). *Machine Bias*. ProPublica. https://www.propublica.org

[13] Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media.

[14] Bogen, D., & Rieke, A. (2018). *Help Wanted: An Examination of Hiring Algorithms, Equity, and Bias*. Upturn.

[15] Jantan, A. R., Hamdan, A. R., & Othman, Z. A. (2009). Knowledge discovery techniques for talent forecasting in human resource application. *World Academy of Science, Engineering and Technology*, *35*, 559–564.

[16] Liem, N., Bartram, D., Reeve, G. W., & Roehling, H. (2018). Modernization of recruitment and selection: The promise of machine learning methods. *Personnel Assessment and Decisions*, *4*(2).

[17] Chen, L., Zhang, Y., & Huang, M. (2021). Resume classification using deep learning and NLP techniques. *International Journal of Computer Applications*, *183*(18), 15–22.

[18] Jura sky, D., & Martin, J. H. (2022). *Speech and Language Processing* (3rd ed.). Stanford University.

[19] Liem, C. C. S., Bartram, D., Reeve, G., & Roehling, M. V. (2018). Modernization of recruitment: The promise of machine learning methods. *Personnel Assessment and Decisions*, *4*(2), 1–12.

[20] Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson.

[21] ird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media.

[22] Jura sky, D., & Martin, J. H. (2022). *Speech and Language Processing* (3rd ed.). Stanford University.

[23] Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

[24] Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

[25] Schreiber, J. (2020). PyMuPDF Documentation. https://pymupdf.readthedocs.io/

[26] Streamlit. (2023). *Streamlit Docs*. https://docs.streamlit.io/