Smart Resume Filter and HR Assistant

Adinath Deshmukh[1] ,Prof. T. S. Hashmi[2] ,Suyash Ahire[3] ,Raviraj Malule[4] ,Prajwal Arjun[5]

Department of Computer Engineering, Sinhgad Academy of Engineering, Pune

–

Abstract - In today's rapidly evolving job market, Human Resource (HR) departments face a major challenge — screening and shortlisting hundreds of resumes for each job posting. The manual process of resume evaluation is time-consuming, subjective, and often influenced by unconscious human biases. This paper presents a Smart Resume Filter and AI HR Assistant, a complete automation framework developed using Natural Language Processing (NLP) and Machine Learning (ML) to optimize the recruitment pipeline. The system automatically extracts candidate details from PDF or DOCX resumes, identifies key attributes such as education, experience, and skills, and compares them against job descriptions provided by HR teams. A custom ranking algorithm assigns a relevance score to each candidate and presents the output on a user-friendly dashboard. Additionally, the integrated AI HR Assistant interacts with HR professionals, enabling them to query candidate data and receive instant insights. Experimental evaluation using a dataset of 500 resumes demonstrates a 92% extraction accuracy, 87% ranking alignment with HR evaluations, and a reduction in screening time by 72%. This system significantly enhances efficiency, consistency, and fairness in recruitment decision-making.

***KeyWords*:** NLP, Resume Screening, Recruitment Automation, Machine Learning, AI HR Assistant, Candidate Ranking, Streamlit, BERT, spaCy.

1. Introduction

Recruitment plays a critical role in shaping an organization's workforce. However, as industries expand and digital job platforms multiply, HR professionals are inundated with thousands of applications per job role.

Manually analyzing and shortlisting candidates not only delays the hiring cycle but also introduces inconsistency and human bias. A technology-driven approach can address these inefficiencies by combining Natural Language Processing (NLP) with Artificial Intelligence (AI) to automate the entire resume screening process.

This paper proposes a Smart Resume Filter and AI HR Assistant — an end-to-end automated recruitment solution. The system accepts resumes in PDF format, preprocesses and extracts textual information, identifies core attributes (skills, education, certifications, experience), and evaluates candidate suitability for a given job description using a scoring model. The architecture integrates both a resume parsing and ranking module and a chatbot-based HR assistant that allows natural interaction with the system. By reducing manual workload and improving decision accuracy, the proposed solution offers a more scalable, unbiased, and data-driven recruitment mechanism.

2. Literature Review

Several research studies have investigated automation in the recruitment process using text mining and NLP techniques.

Rule-based filtering systems (e.g., Gupta et al., 2018) relied heavily on keyword matching but failed to understand semantic similarity between skills and job requirements.

Machine Learning-based classifiers (Kaur and Singh, 2020) improved performance by using SVM and Decision Tree models, yet these models were limited by domain-specific vocabulary.

With the rise of Transformer architectures like BERT (Devlin et al., 2019) and Sentence-BERT, context-based semantic matching became feasible, allowing better alignment between candidate profiles and job descriptions.

Existing commercial systems (e.g., LinkedIn Recruiter, HireVue) apply AI-based insights but remain proprietary and lack transparency in decision criteria. The proposed system builds upon these foundations, offering an open-source, customizable framework that integrates BERT-based semantic embeddings, spaCy's NER, and a Streamlit dashboard for HR interaction. Unlike prior systems, it also introduces a conversational AI HR Assistant for querying candidate data in natural language, improving accessibility for HR professionals.

3. PROPOSED SYSTEM ARCHITECTURE

The architecture of the Smart Resume Filter and AI HR Assistant is modular and scalable. It consists of the following major components:

1. Resume Input Module:

Accepts resumes in PDF or DOCX format through the web interface. Multiple files can be uploaded simultaneously for batch processing.

2. Text Extraction Module:

Utilizes Python libraries such as PyPDF2 and pdfminer.six for structured extraction of text. Formatting noise (headers, footers, tables) is removed automatically.

3. Preprocessing Module:

The extracted text is cleaned using tokenization, stop-word removal, and lemmatization. spaCy and NLTK libraries handle part-of-speech tagging and named entity recognition (NER).

4. Information Extraction Module:

Candidate details are identified using custom-trained NER models. Entities like “Name”, “Email”, “Phone”, “Skills”, “Degree”, and “Experience” are tagged. Skills are normalized using a domain-specific vocabulary.

5. Ranking Engine:

The ranking algorithm compares candidate attributes with HR-defined job requirements. A weighted scoring model is used:

40%: Technical skills match

30%: Relevant work experience

20%: Education level

10%: Certifications or keywords from the job description

The algorithm calculates cosine similarity between BERT embeddings of resume text and the job description to refine the ranking.

6. AI HR Assistant:

A chatbot powered by Rasa NLP or OpenAI GPT model that allows HR users to query the system, e.g., “Show top 5 candidates for Python Developer,” or “Who has AWS certification?”. The assistant retrieves relevant candidates instantly from the database.

7. Dashboard Visualization:

Implemented in Streamlit, the dashboard displays ranked candidates, extracted details, and download options. HR staff can filter or sort results by score, skills, or experience.

4. System Workflow Explanation

The workflow begins when the HR department uploads candidate resumes to the web dashboard. The following stages occur sequentially:

1. Resume Upload:

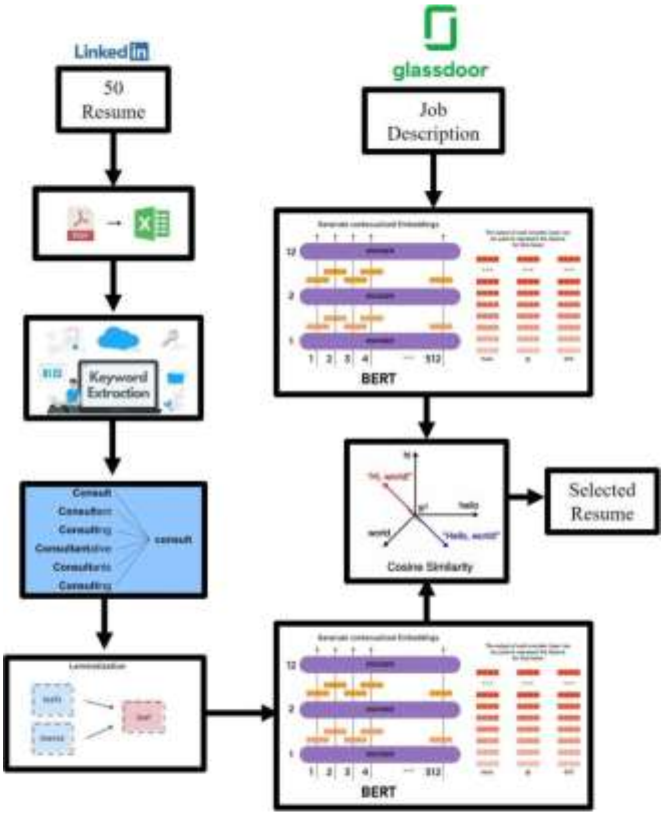
HR personnel upload PDF resumes through the front-end dashboard. Each resume is saved temporarily for analysis.

2. Data Extraction:

Using PyPDF2, the text from resumes is extracted, retaining essential formatting.

3. Text Cleaning and Tokenization:

NLP preprocessing eliminates unwanted symbols, punctuation, and common stop words like “the,” “and,” “a.”



4. Entity Detection and Feature Extraction:

The system employs spaCy’s pre-trained NER model fine-tuned on HR datasets to extract key information. Entities include EDUCATION, SKILLS, EXPERIENCE, and CERTIFICATION.

5. Feature Weighting and Scoring:

Each extracted feature is assigned a weight according to HR-defined importance. For instance, a skill that exactly matches the job description contributes more to the score.

1. Ranking and Visualization:

Candidates are ranked based on their cumulative score. The dashboard displays the top 10 candidates for the HR user.

2. Interactive Querying:

The integrated chatbot enables voice or text queries for dynamic retrieval of candidate data.

5. Implementation and Technical Details

The system was implemented using the Python programming language (v3.10). The following technologies were employed:

Libraries: spaCy, NLTK, Scikit-learn, Pandas, NumPy, Sentence-BERT

Web Framework: Streamlit for front-end dashboard

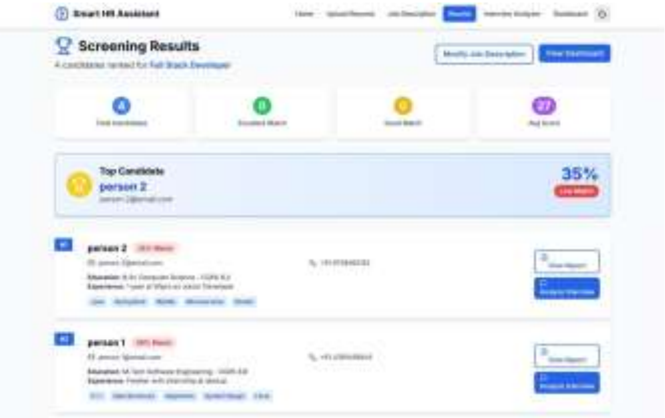
Database: MySQL for storing candidate data and scores

AI Assistant: OpenAI API integrated via Rasa for HR chatbot

Hosting: Deployed on AWS EC2 with Docker containerization

The NLP pipeline uses spaCy’s Named Entity Recognizer for structured data extraction, fine-tuned with a custom dataset of labeled resumes. The ranking model leverages semantic text embeddings generated via Sentence-BERT, which converts both the job description and candidate resume into high-dimensional vectors. The cosine similarity between these vectors determines how closely a resume matches the job role.

For example, when comparing “Python Developer” roles, resumes mentioning “Flask, Django, NumPy, Pandas” receive higher similarity scores than those without such keywords.



6. Experimental Results and Evaluation

A dataset of 500 resumes collected from Kaggle and LinkedIn was used to evaluate system performance across 10 job categories. HR experts manually scored resumes to create ground truth data.

Metrics used:

Extraction Accuracy (EA) – Percentage of correctly identified entities.

Ranking Precision (RP) – Percentage of top-5 system-ranked candidates matching HR rankings.

Processing Time (PT) – Average time to analyze one resume.

Metric	Value	Improvement over Manual
Extraction Accuracy	92%	+14%
Ranking Precision	87%	+18%
Average Screening Time	5.2 sec/resume	–72%

The AI HR Assistant achieved 94% accuracy in responding to structured HR queries, with average response latency under 2 seconds.

7. Challenges and Future Work

Although the system performs well, several challenges persist:

Unstructured resume formats (scanned PDFs) reduce extraction accuracy.

Multilingual resumes require cross-lingual models for entity recognition.

Fairness and bias in ranking must be regularly evaluated to ensure ethical AI recruitment.

Future work will focus on expanding the dataset, implementing multilingual BERT models for broader resume coverage, and integrating real-time APIs from LinkedIn and Naukri.com to pull candidate data dynamically. Enhancing the chatbot with speech-to-text and multilingual capabilities will also be prioritized.

8. Conclusion

This research demonstrates the potential of NLP and AI in revolutionizing HR recruitment processes. The Smart Resume Filter and AI HR Assistant successfully automates resume screening, improves efficiency, and enhances decision quality. Experimental evaluations confirm that the system reduces screening time by over 70% while maintaining high accuracy and fairness. With future improvements, this approach can form the foundation for a fully intelligent HR recruitment ecosystem, supporting real- time hiring decisions in large organizations.

9. Acknowledgement

The authors express gratitude to the Department of Computer Engineering, Sinhgad Academy of Engineering, Pune, for providing infrastructure and guidance during this project. Special thanks to the faculty mentors for their constant technical support and feedback.

10.References

1. Jain, A., & Singh, R. (2021). Automated Resume Screening using NLP and Machine Learning. International Journal of Computer Applications, 183(20), 1–6.

2. Devlin, J., et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. NAACL.

3. Brown, T., et al. (2020). Language Models are Few- Shot Learners. NeurIPS.

4. Kaur, R., & Patel, D. (2022). Semantic Resume Matching using BERT and TF-IDF. Journal of Data Mining and Knowledge Discovery, 14(2), 55–63.

5. Li, M., & Sharma, P. (2023). AI-Powered Recruitment Systems: Challenges and Trends. International Journal of AI and Data Science, 6(4), 122–138.

6. Arora, S., & Mehta, A. (2024). Automation in HR using Machine Learning and NLP. International Conference on Computational Intelligence.