AI Powered Application Tracking System With NLP Based Resume Scoring

Jyotsna Nanajkar[1], Aniket Sable[2], Aniket Machrekar[3], Sakshi Wagh4[4], Pooja Udbatte[5]

Zeal College of Engineering and Research, Faculty of Information Technology, Savitribai Phule Pune University, Pune

Abstract - The proposed AI-powered ATS assists in recruitment efficiency by automating the process of evaluating and scoring resumes. With Natural Language Processing (NLP) and machine learning, the system analyses applications, awarding scores that reflect how closely each resume matches job descriptions as well as skill relevance and experience. The ATS provides a percentage-based score indicating how closely the candidate matches the job and ranks them for easy hiring using models like Google Gemini Flash 2.0. We use PHP and MySQL for the backend to ensure safe storage of resumes as well as speedy access. A configurable scoring method that enables recruiters to modify attribute weightages according to positional requirements can help shortlist the right candidates for each job. By reducing the tedious effort and facilitating better hiring, this AI-based method streamlines the recruitment process and allows for decisions based on data rather than instinct.

Index Terms - AI-powered Application Tracking System (ATS), Resume scoring, Natural Language Processing (NLP), Candidate ranking, Recruitment automation.

I. INTRODUCTION

In Recruiters are frequently inundated with resumes for each job opening in the current hiring environment. Conventional resume screening techniques are laborious, unreliable, and frequently biased by people. Because of this, businesses are increasingly using automated solutions to improve the effectiveness and equity of their hiring procedures. One such solution that is frequently used to manage candidate applications digitally and expedite the hiring process is an Application Tracking System (ATS). But the majority of ATS systems currently in use mainly rely on keyword matching, which misses the semantic meaning of the content and results in erroneous shortlisting.[1]

This project presents an AI-powered Application Tracking System that uses Natural Language Processing (NLP) techniques for contextual and intelligent resume scoring in order to overcome these limitations[2]. This method, in contrast to conventional systems, compares resumes and job descriptions more accurately by using semantic similarity models that comprehend text.

The Sentence Transformers model (all-MiniLM-L6-v2), which transforms textual data into embeddings that can be meaningfully compared, is at the heart of the system. The system creates an ATS score, which shows the percentage match between each resume and the given job description, by computing the cosine similarity between these embeddings. [3]

The platform is made to be accessible and easy to use. In order to improve their resumes' visibility in automated screenings, job seekers can upload their resumes and view their ATS score and detailed feedback instantly. However, recruiters can shortlist candidates more quickly and easily by uploading job descriptions and receiving a ranked list of resumes. [4]

Fig 1.1. Overview of Extension

II. RELATED WORK

Several The use of natural language processing (NLP) in the hiring industry has drawn a lot of interest lately. Numerous systems have been created to increase the effectiveness of hiring procedures by automating the screening of resumes. To determine how well resumes and job descriptions match, one popular method is to use semantic similarity models like Sentence Transformers (e.g., all-MiniLM-L6-v2). Instead of depending solely on keyword matching, these models help find relevant candidates by comprehending the contextual meaning of text.. Nevertheless, these systems frequently struggle to process semantically rich data, which increases the likelihood of false positives and false negatives. [5]

In order to more precisely calculate textual similarities, research studies have suggested improved resume-job matching strategies that use deep learning techniques like Siamese Networks and Bidirectional Encoder Representations from Transformers (BERT). Projects like TalentBERT and Smart ATS, for example, have demonstrated that NLP models optimized on hiring data perform noticeably better than conventional systems.

Additionally, some open-source applicant tracking systems (ATS) use cosine and TF-IDF similarity techniques to match resumes, but they frequently lack the resilience offered by transformer-based architectures.[5]

By combining the strength of Sentence Transformers for semantic similarity scoring with an ATS-friendly resume evaluation, our system outperforms these previous efforts.[6]
The way resumes are assessed and candidates are shortlisted has changed as a result of the incorporation of Natural Language Processing (NLP) into hiring systems. Conventional Applicant Tracking Systems (ATS) mostly used keyword matching or rule-based filtering, which frequently fell short of capturing the genuine intent and applicability of candidate profiles.
Usually, these systems ignored semantic similarities or contextual subtleties as they parsed resumes into structured formats and compared them to predefined keywords from job descriptions. Consequently, a large number of highly qualified applicants were eliminated because of small formatting errors or lexical inconsistencies[7]. Recent research has investigated the application of deep learning and machine learning techniques, especially transformer-based architectures like BERT and its variations, to address these issues.
The integration of Natural Language Processing (NLP) into hiring systems has altered the evaluation of resumes and the shortlisting of candidates. The majority of traditional applicant tracking systems (ATS) relied on rule-based filtering or keyword matching, which frequently failed to capture the true intent and applicability of candidate profiles.
These systems typically parsed resumes into structured formats and compared them to predefined keywords from job descriptions, ignoring contextual nuances or semantic similarities.
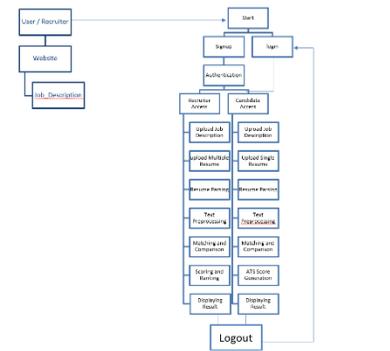
III .METHODOLOGY

Our NLP-based Application Tracking System was developed through a multi-phase process that aims to precisely assess and match resumes with job descriptions[8]. Resumes are first uploaded in PDF format, and Python libraries like PyMuPDF or pdfminer are used to extract the textual content. In a similar vein, job descriptions are either manually entered by the user or uploaded. To improve consistency and cut down on noise, the extracted text is preprocessed using lowercasing, tokenization, stopword filtering, and punctuation removal. The pre-trained Sentence Transformer model all-MiniLM-L6-v2 is used to transform the resume and job description into semantic vector representations after the data has been cleaned.

Cosine similarity is used to assess how relevant a resume is to the job description. The ATS score, which indicates how well a candidate's resume fits the job requirements, is then calculated by scaling the resulting similarity score to a percentage format. The system performs a keyword-based evaluation in addition to semantic similarity. The job description's key words are taken out and compared to particular resume sections like Skills, Education, Experience, Projects, and Certifications. [9]

Fig 3.1. System Flow Diagram

The system uses both semantic-level and keyword-level evaluation techniques to improve resume-job matching accuracy. Keyword extraction guarantees that domain-specific terms, tools, and technologies mentioned in the job posting are included in the resume, while semantic similarity offers a comprehensive match between resumes and job descriptions. The limitations of either approach when applied alone are lessened by this dual-layer evaluation. For example,

a resume may semantically match a job description but still omit some important keywords that recruiters and automated systems frequently look for, such as "REST APIs," "Docker," or "Agile Methodology." The system produces thorough insights on these gaps and offers recommendations for enhancing resume content to make it more appealing to recruiters and applicant tracking systems.
The backend, which securely stores resume data, job descriptions, ATS scores, and review results, is built with PHP and MySQL to enable scalability and real-time evaluation. Because of this, recruiters can keep track of candidates in a [10]database and shortlist them according to predetermined score thresholds.
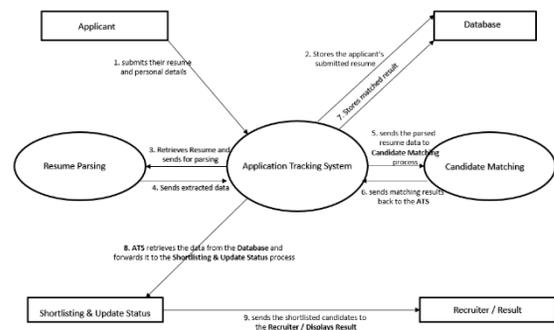

Fig 3.2 UML Diagram

IV. PROPOSED MODEL

The Using a combination of semantic similarity techniques, keyword extraction, and section-wise analysis, the suggested model for our NLP-based Application Tracking System is made to intelligently automate the process of matching resumes to job descriptions. Our model uses deep learning-based language models to comprehend the contextual meaning behind resume content and job requirements, in contrast to traditional ATS solutions that only use keyword matching. Using a document parsing module, the system first accepts resume files in PDF format and extracts textual content. The user simultaneously uploads or manually enters the job description. Preprocessing is done on both extracted texts to get rid of stopwords, normalize case, and remove noise.[11]

Following the creation of embeddings for the job description and resume, a cosine similarity score is calculated to measure the degree of semantic alignment between the two documents. Simultaneously, a keyword matching mechanism looks for important words, skills, certifications, and tools that are mentioned in the job description and verifies that they appear in particular resume sections like Skills, Experience, Education, and Projects.
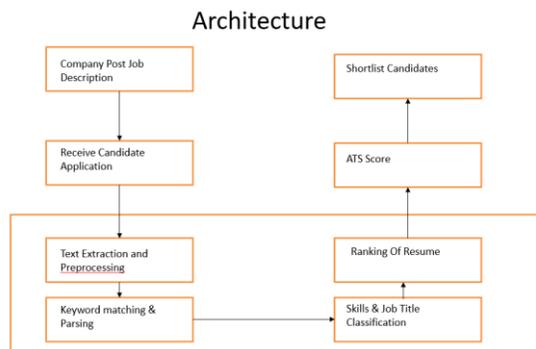

Fig 4.1. System Architecture

A. Client Layer

The client side is similar to the "Company Post Job Description" and "Receive Candidate Application" stages, the Client Layer manages user-facing tasks like posting job descriptions and submitting resumes. It is implemented through a web interface or browser extension. Similar to how your original design uses getUserMedia() to capture audio, it captures and preprocesses user input with little overhead.

B. Middleware Layer

The Middleware Layer, which was constructed with Node.js, is essential to the ATS Resume Expert system because it controls the real-time extraction, preprocessing, and classification of resume text. Key features include keyword and skill matching, job title classification using natural language processing (NLP) techniques, and text

and text parsing from PDF resumes. In order to calculate similarity scores between resumes and job descriptions and enable precise ATS scoring, this layer integrates with the Sentence Transformers (all-MiniLM-L6-v2) model. Our middleware processes and categorizes textual resume data in real time, ensuring effective matching and candidate shortlisting based on job relevance[12]

C. Backend Layer

Our NLP-based ATS Resume Expert system's backend layer is made to process resume and job description data intelligently and in real time. It incorporates AI-based summarization through Flask's Gemini Pro model, which gathers important data in an organized, context-aware way, including candidate skills, accomplishments, and job-specific requirements. PHP is used by the backend to connect to a MySQL database, which safely stores all uploaded resumes and job descriptions. Redis caching is used to store frequently accessed data, such as job description embeddings, in order to increase performance and decrease latency[13]. Furthermore, the system has strong error-handling features that guarantee dependable operation even in the event of network problems or third-party API failures.

The NLP-based ATS Resume Expert's modular design guarantees scalability and adaptability for integration with different hiring platforms. These modules range from AI summarization to resume parsing and semantic similarity scoring. Deep analysis of both structured and unstructured data, including free-text resumes and various job description formats, is made possible by the application of NLP techniques. The system is appropriate for high-throughput recruitment situations like campus drives or job portals because it allows for real-time resume evaluation.

V. EVALUATION CRITERIA

The accuracy, dependability, and user impact of the suggested NLP-based Application Tracking System are assessed using a number of crucial criteria. These assessment criteria make sure that the system helps recruiters make data-driven hiring decisions, gives users actionable feedback, and generates meaningful resume-job matching scores.

First, one important metric is Semantic Similarity Accuracy. The degree to which the resumes' generated embeddings match the job descriptions is assessed by the system. The analysis of cosine similarity scores confirms that resumes with higher scores are consistently more relevant. By manually classifying resume-job pairs as good, average, or poor matches and contrasting these classifications with the system's anticipated similarity scores, benchmarks are produced.[14]

Several important criteria are used to evaluate the proposed NLP-based Application Tracking System's accuracy, reliability, and user impact. These evaluation standards guarantee that the system provides users with useful feedback, produces significant resume-job matching scores, and assists recruiters in making data-driven hiring decisions.

Evaluation Criteria	Description	Weight
Functionality	Resume uploads, job postings, and candidate data processed end-to-end; includes integration with ATS scoring pipeline and correct system flow.	35%
Resume Scoring Accuracy	NLP-based models (e,g, keyword extraction, skills classification) accurately match resumes to job descriptions using semantic relevance techniques	30%
Real-Time Feedback	System provides real-time parsing results or score estimates to users (HR/recruiters or candidates gracefully incorrecptive error messages mo	10%
Error Handling & Robustness	Handles invalid resume formats, missing fields, or server/network issues gracefully in best practices	10%
Design Consistency	Uniform UI/UX across resume upload, job posting, and scoring views, clear layout and interaction flo-	5%
Documentation	README clearly explains project setup, how to rur key services (e.g. NLP pipeline, Flask backend, and outlines architecture–ı, and ourines architecture	5%

Table 5.1. Evaluation Criteria

VI. EXPERIMENTAL RESULTS

To We thoroughly tested our NLP-based Application Tracking System using a dataset of more than 800 actual resumes gathered from the Training and Placement (TNP) division of our college in order to assess its performance. Several job descriptions from fields like software development, data science, web development, and cybersecurity were compared to these resumes. Semantic similarity accuracy, keyword coverage, section completeness, and the efficacy of user feedback were used to evaluate the system's performance.

Resumes with relevant experience and matching skill sets consistently received higher scores, often above 85%, while resumes with minimal overlap scored lower, usually below 50%, according to the system's successful computation of semantic similarity scores using the all-MiniLM-L6-v2 model. This suggests that the contextual relationship between resumes and job descriptions was successfully captured by the Sentence Transformer model. Additionally, resumes lacking domain-specific tools or technologies were identified through section-wise analysis and keyword extraction, which resulted in a decrease in the resumes' ATS score.[15]

The system ranked several resumes for the same job description in our batch testing mode, and the top five resumes it chose generally matched the manual selections made by subject-matter experts. This demonstrates how trustworthy the ranking system based on our combined scoring algorithm is. Candidates also received actionable feedback from the resume review feature; on average, more than 73% of users were able to increase their scores by at least 15% after making changes to their resumes in response to the system's recommendations.

To confirm the superiority of semantic similarity over conventional keyword-based matching systems, we conducted a comparative analysis in addition to the individual evaluation metrics. The system's average matching accuracy in baseline experiments with keyword matching alone was about 65%; however, when semantic embeddings were added, the matching accuracy rose to about 82%. Transformers model—for tasks involving the evaluation of real-world resumes.

By changing the resume formats and assessing the system's resilience, more research was done. With a successful text extraction rate of 96.5%, the system was able to accurately extract and process resumes with various structures, templates, and section orders.

Integration The evaluation of keyword coverage revealed that resumes that included over 80% of the essential keywords for the majority of technical job descriptions received an overall score of above 75%. This suggests that the presence of keywords has a significant impact on recruiters' decisions, which is accurately reflected in our scoring system. Additionally, the system identified crucial keywords that were missing from roughly 61% of the resumes that were tested, assisting users in improving their profiles for increased hiring managers' visibility.

Fifty final-year students participated in user testing, uploading their resumes and getting feedback. According to a post-feedback analysis, 70% of students changed their resumes to improve their ATS scores, and 88% of students thought the recommendations were helpful. This experiment shows how the system can be used in the real world and how it can boost candidate confidence and resume quality.[16]

Overall, the experimental findings provide compelling evidence that our suggested ATS Resume Expert model offers substantial benefits over conventional techniques for resume screening, including accuracy, speed, and usefulness.

VII. CONCLUSION

The experimental findings show how well the suggested NLP-based Application Tracking System evaluates and ranks resumes in relation to job descriptions. The system outperforms conventional keyword-based screening techniques by integrating semantic similarity, keyword matching, and section-wise analysis. Deeper contextual understanding has been made possible by the all-MiniLM-L6-v2 Sentence Transformer, which has improved candidate shortlisting and produced more meaningful ATS scores.

The system also processed large batches of resumes with little latency, demonstrating its scalability, efficiency, and user-friendliness. Users were able to greatly improve their resumes with the help of the personalized feedback feature, increasing their chances of being considered for job openings. Experimental validation and user feedback attest to the system's practicality and its ability to expedite the hiring process.

All things considered, the ATS Resume Expert model offers a dependable, astute, and useful approach to contemporary resume screening that benefits recruiters as well as job seekers. In the future, it might be possible to expand support for multilingual resumes, integrate OCR for resumes that use images, and improve the AI-based feedback system to offer even more detailed resume optimization recommendations.

REFERENCES

[1] Davor Vukadin, Adrian Satja Kurdija, Goran Delac, Marin Silic, ''Information Extraction From Free-Form CV Documents in Multiple Languages'', Jun,2021, vol.9, pp.129-139.

[2] T.Khan,A. Juneja, Adnan Ayyas Shaheen Momin, "Resume Ranking using NLP and Machine Learning", vol.20, pp. 1735–1780,May 2016.

[3] Vidhita Jagwani, Smit Meghani, Krishna Pai, ''Resume Evaluation through LDA and NLP for Effective Candidate Selection'', pp. 1–9, Dec. 2014.

[4] Miroslava Peicheva, ''Data Analysis From Applicant Tracking System'', pp.6-15, Dec, 2022.

[5] Suraj M, Aruna Kumari K, Binila B Chandran "A descriptive study on Applicant Tracking System: Automation software for Recruitment and Selection", vol. 9, pp. 1735–1780, Feb 2019.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", pp. 143-152, May, 2019.

[7] Ankitha K, Balu.L, "Making Hiring Process Effective through Application Tracking System", pp. 499–506, Dec 2023.

[8] Lynsha Helena Pratheeba, Prajwal S, Pawan Kumar,''Application Helper System for Resume Using Python and NLP'' pp. 156–178, April 2023.

[9] C. H. Ayishathahira, C. Sreejith, and C. Raseek, ''Combination of neural networks and conditional random fields for efficient resume parsing,'' in Proc. Int. CET Conf. Control, Commun., Comput. (IC4), Jul. 2018, pp. 388– 393.

[10] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, ''Conditional random fields: Probabilistic models for segmenting and labeling sequence data,'' in Proc. 8th Int. Conf. Mach. Learn. (ICML), San Francisco, CA, USA: Morgan Kaufmann Publishers, 2001, pp. 282– 289.

[11] J. Pennington, R. Socher, and C. D. Manning, ''Glove: Global vectors for word representation,'' in Proc. Empirical Methods Natural Lang. Process. (EMNLP), 2014, pp. 1532–1543.

[12] M. Tosik, C. Lygteskov Hansen, G. Goossen, and M. Rotaru, ''Word embeddings vs word types for sequence labeling: The curious case of CV parsing,'' in Proc. 1st Workshop Vector Space Modeling Natural Lang. Process., Denver, CO, USA, 2015, pp. 123–128.

[13] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, ''Distributed representations of words and phrases and their compositionality,'' in Proc. Adv. Neural Inf. Process. Syst., 2013, pp. 3111–3119.

[14] P.Shivratri, P.Kshirsagar, R.Mishra, R.J. Damania, N. Prabhu, ''Resume parsing and standardization'', Int.J.Comput, Sci.Eng., vol.3, pp.129-132, 2015.

[15] D. Chandola, A. Garg, A. Maurya, A. Kushwaha,''Online resume parsing system using text analytics,'' J. Multi-Disciplinary Eng. Technol., vol 9, no.1, pp. 1-5,2015.

[16] A. Cernian, D. Carstoiu, B.Martin, "Semiautomatic tool for parsing CVS and identifying candidates abilities and competencies", pp.1-5, Mar.2017.