



AJEENKYA
D Y PATIL UNIVERSITY
THE INNOVATION UNIVERSITY

School of
Engineering

ApplyXpert

Project submitted in partial fulfillment
of the requirements of the degree of

Bachelor of Technology

in

Artificial Intelligence & Data Science

by

Devashish Sharma

2022-B-12092003

Under the Supervision of

Dr. Ajitkumar Pundge



AJEENKYA
D Y PATIL UNIVERSITY
THE INNOVATION
UNIVERSITY

Apr 2025

School of Engineering

Ajeenkya DY Patil University, Pune



AJEENKYA
D Y PATIL UNIVERSITY
THE INNOVATION UNIVERSITY

School of
Engineering

Apr 23, 2025

CERTIFICATE

This is to certify that the dissertation entitled “**ApplyXpert**” is a bonafide work of “**Devashish Sharma (2022-B-12092003)**”, submitted to the School of Engineering, Ajeenkya D Y Patil University, Pune in partial fulfillment of the requirement for the award of the degree of “**Bachelor of Technology in Artificial Intelligence & Data Science**”.

Dr. -----

Supervisor

Acknowledgement

I remain immensely obliged to Dr. Ajitkumar Pundge, for providing me with the idea of this topic, and for her invaluable support in garnering resources for me either by way of information or computers also his guidance and supervision which made this Project happen.

I would like to say that it has indeed been a fulfilling experience for working out this Project.

Devashish Sharma

ApplyXpert

April 23, 2025

Abstract

ApplyXpert is a smart and user-friendly platform that helps job seekers improve their resumes by matching them with job descriptions using AI and NLP. It analyzes resumes for relevant skills, experience, and keywords, then provides a score and personalized feedback to increase their chances of passing ATS filters. With features like multi-resume analysis, skill gap detection, and report generation, ApplyXpert simplifies the job application process. Built with modern web technologies and machine learning, it's designed to make resume optimization easy, efficient, and impactful.

Index Terms: Resume Scoring, NLP, ATS, Skill Matching, AI, Job Application, Resume Analysis, Career Tools.

1 Introduction

In today's competitive job market, standing out among hundreds of applicants is no easy feat. Most companies rely on **Applicant Tracking Systems (ATS)** to filter resumes before they ever reach a human recruiter. These systems scan resumes for relevance based on job descriptions, often disqualifying qualified candidates due to formatting issues or keyword mismatches. This process can be discouraging for job seekers who are unaware of how ATS works or how to tailor their resumes accordingly. The **ATS Resume Analyzer** is developed as a smart solution to bridge this gap. By leveraging **Natural Language Processing (NLP)** and **Machine Learning (ML)**, the platform evaluates resumes against job descriptions to generate a match score, identify missing skills, and offer personalized improvement suggestions. It allows users to upload multiple resumes, compare them against various job roles, and even generate detailed PDF reports with actionable feedback. This system is designed not just as a tool, but as a guide to help users create more optimized, job-ready resumes that have a better chance of making it through ATS filters.

1.1 Problem Statement

Job seekers often face rejection not because of a lack of qualifications, but due to poorly optimized resumes that do not meet ATS requirements. Traditional methods of resume review are time-consuming, inconsistent, and offer limited feedback. There's a need for an intelligent system that provides fast, personalized, and actionable insights to improve resume effectiveness.

1.2 Scope of the Project

ApplyXpert is designed for job seekers across industries looking to optimize their resumes for better job matching. The system supports real-time resume scoring, multi-resume comparisons, and report generation. It is scalable, adaptable for various domains, and can be enhanced in the future with advanced ML models, real-time feedback tools, and job board integration.

1.3 Contribution

This project introduces a web-based ATS Resume Analyzer that uses NLP and machine learning to evaluate resumes against job descriptions. Key contributions include:

- Implemented resume parsing and semantic similarity scoring using spaCy.

- Developed a React.js frontend for an intuitive user interface.
- Built a Node.js backend to handle API requests and integrate with Python-based ML models.
- Enabled multi-resume comparison and generated detailed PDF reports with actionable feedback.
- Designed a custom ATS scoring algorithm combining keyword match, domain relevance, and experience evaluation.

2 System Overview

2.1 Features

- Resume analysis with ATS score generation.
- Job description matching using NLP.
- Skill gap detection and improvement suggestions.
- PDF report with detailed feedback.

2.2 Advantages over Existing Systems

- Uses semantic analysis, not just keyword matching.
- Provides clear, personalized feedback.
- Simple, fast, and user-friendly interface.

2.3 Stakeholders

- Job Seekers.
- Career Coaches.
- Recruiters.

3 Background

In today's competitive job market, many candidates struggle to get noticed by employers due to automated resume screening tools, also known as Applicant Tracking Systems (ATS). Most resumes fail to pass these filters because they lack alignment with the job description. ApplyXpert aims to bridge this gap by using AI and NLP to analyze resumes and suggest improvements that enhance visibility and relevance.

3.1 Literature Review

Applicant Tracking Systems (ATS) are now standard in modern recruitment, helping employers manage the overwhelming volume of job applications. However, many well-qualified candidates are filtered out early because their resumes don't align with ATS keyword requirements or formatting standards. This has led to the development of resume optimization tools aimed at increasing the

chances of passing these automated filters.

Natural Language Processing (NLP) plays a key role in improving resume analysis. Open-source libraries like spaCy offer efficient capabilities for named entity recognition, keyword extraction, and semantic similarity—all of which help assess how well a resume matches a job description. Combining these techniques with machine learning enables smarter, context-aware analysis rather than just keyword matching.

While platforms like Jobscan provide basic resume-to-job description matching, they often lack flexibility and depth in their analysis. In contrast, research-backed approaches using models like BERT offer deeper contextual understanding, enabling more accurate and personalized resume evaluations.

This project leverages proven NLP methods and scoring algorithms to create a practical and accessible resume analyzer. By offering multi-resume comparison, domain-based skill extraction, and PDF report generation, it aims to bridge the gap between applicant capabilities and ATS expectations—ultimately empowering job seekers with better insights and tailored recommendations.

4 System Design and Development

4.1 System Architecture

The ATS Resume Analyzer is built on a modular, service-oriented architecture to ensure scalability, maintainability, and high performance. The system adopts a three-tier structure comprising the Presentation Layer (Frontend), Application Layer (Backend API And ML Logic), and Data Layer (Database and Reports).

4.1.1 Architecture Overview:

- **Frontend:** A single-page application (SPA) developed using React.js with Material UI components. It handles user interactions such as file uploads, score visualization, and result downloads.
- **Backend:** Built using Node.js with Express.js, it serves as a RESTful API gateway, handling client requests, invoking the ML layer, and managing communication with the database and file system.
- **Machine Learning Module:** Implemented in Python, utilizing spaCy's NLP capabilities to perform text extraction, semantic similarity scoring, keyword extraction, and resume-job relevance classification.
- **Database Layer:** MongoDB serves as the NoSQL document-based database, managing user inputs, job descriptions, resumes, and analysis outcomes.
- **Reporting:** PDF reports are generated using FPDF and PyMuPDF libraries and stored for user access through secure endpoints.

4.2 Technology Stack

- **Frontend:** React.js, Material UI, Axios, React Router DOM
- **Backend:** Node.js, Express.js, Cors, Multer, Dotenv
- **Machine Learning:** Python, spaCy, scikit-learn, NumPy, pandas
- **Database:** MongoDB, Mongoose
- **Reporting:** FPDF, PyMuPDF (fitz)
- **Version Control:** Git, GitHub
- **Authentication & Security:** JSON Web Tokens (JWT), Bcrypt, Helmet, HTTPS

4.3 Modules and Workflow

- **1. Resume Upload Module:** Allows users to upload their resumes (PDF format) via the frontend interface. The files are sent to the backend using Axios and stored temporarily for processing.
- **Job Description Input Module:** Users can input one or multiple job descriptions, either by uploading a file or typing directly into a text field.
- **NLP Processing Module:** The backend invokes the Python ML layer which uses spaCy and scikit-learn to extract keywords, skills, and experience from both resumes and job descriptions. It computes semantic similarity scores and performs skill gap analysis.
- **Scoring Module:** Generates a compatibility score between resume and job description using NLP-based metrics. Results are returned in JSON format to the frontend.
- **Report Generation Module:** Generates downloadable PDF reports using FPDF and PyMuPDF. These reports include resume score, missing skills, and suggestions.
- **User Interface Module:** Built with React.js and Material UI, this module displays results, visualizations (charts, match scores), and handles user interactions like navigation and downloads.
- **Database Module:** Stores resumes, job descriptions, scores, and reports in MongoDB for future retrieval and analysis. Uses Mongoose for schema modeling.
- **Security Module:** Ensures user data is protected using JWT for authentication, Bcrypt for password encryption, and secure HTTPS communication.

5 Implementation

5.1 Front-end Implementation

The frontend of the ATS Resume Analyzer is built using React.js, leveraging Material UI for styling and component design to deliver a clean and intuitive user experience. The application follows a component-driven architecture, which ensures modularity, re-usability, and maintainability.

5.1.1 Key Functionalities:

- **File Upload Interface:** Users can upload resumes (PDF) and job descriptions. The UI provides real-time validation for supported formats and file size limits.
- **ATS Score Visualization:** Once analysis is complete, the UI displays results using progress bars, pie charts, and badges, giving a clear picture of overall alignment, keyword match, and domain relevance.
- **Skill And Keyword Analysis Display:** Highlights extracted skills from the resume and compares them with those in the job description. Missing or weak keywords are also presented to help users focus on specific improvements.
- **Report Download Section:** Users can download a PDF report of their analysis, which includes scores, recommendations, and detailed breakdowns.

5.2 Back-end Implementation

The backend is developed using Node.js with Express.js, providing a robust and scalable foundation for RESTful API services. It

is responsible for orchestrating file processing, interacting with the ML engine, managing data persistence, and handling user interactions.

5.2.1 Core Responsibilities:

- **File Handling and Preprocessing:** i. Uploaded documents are validated and passed to the Python module via system calls or inter-process communication. Files are temporarily stored for analysis and deleted post-processing if not needed.
- **API Services:** ii. The backend exposes multiple endpoints: 1./api/ats/analyze: Accepts resumes and job descriptions for comparison. 2./api/ats/score: Returns individual and composite scores. 3./api/ats/missing: Provides a list of missing or underrepresented keywords. 4./api/ats/reports/:id: Serves generated PDF reports for download.
- **Integration with ML Layer:** The backend communicates with Python-based scripts (using `child_process` or `spawn`) that run NLP operations via `spaCy`. Results are parsed and returned to the frontend.
- **Security And Validation:** Express middleware handles input sanitization, rate limiting, and file type validation to ensure application integrity.

5.3 Machine Learning Implementation

5.3.1 NLP Framework and Library The core of the ATS Resume Analyzer leverages the `spaCy` library in Python, using the `en_core_web_sm` model for natural language processing. It processes both resumes and job descriptions by tokenizing, lemmatizing, and extracting linguistic features like named entities and part-of-speech tags.

5.3.2 Text Parsing and Preprocessing PDF files are parsed using `PyMuPDF` (`fitz`), and the extracted raw text is cleaned by removing special characters, stopwords, and unnecessary formatting. This ensures consistent and high-quality input for downstream NLP tasks.

5.3.3 Resume Section Detection Custom rule-based patterns and keyword anchors are used to detect and segment standard resume sections such as “Skills,” “Experience,” “Projects,” and “Education.” This helps target relevant areas for skill extraction and scoring.

5.3.4 Skill and Keyword Extraction Skills are extracted from resumes using a pre-defined taxonomy covering technical and domain-specific categories. The extracted skills are compared against those in the job description to calculate a match percentage.

5.3.5 Semantic Similarity Calculation The similarity between a resume and a job description is measured using `spaCy`'s word vector-based semantic similarity. This evaluates contextual relevance rather than just keyword overlap.

5.3.6 Experience and Domain Relevance Scoring The system extracts job roles, durations, and relevant tools from the experience section of the resume and compares them with expected experience in the job description. Additionally, domain-specific

keywords are used to evaluate relevance to fields like Finance, Cybersecurity, or Healthcare.

5.3.7 Scoring Algorithm Multiple scoring components are combined to generate a final composite score:

- **Semantic similarity score (25%):** Measures contextual relevance between resume and job description.
- **Keyword match score (25%):** Based on direct skill and keyword overlap.
- **Experience relevance score (20%):** Analyzes job roles, durations, and related tools.
- **Domain alignment score (15%):** Focuses on sector-specific terminology (e.g., Cybersecurity, Healthcare).
- **Resume structure and formatting score (15%):** Evaluates readability and ATS-friendliness.

5.3.8 Skill Taxonomy and Knowledge Base The system uses structured dictionaries for technology skills (e.g., programming languages, frontend/backend tools, databases, cloud platforms) and domain skills (e.g., healthcare, finance, cybersecurity). These dictionaries improve keyword classification and provide context-aware analysis.

5.3.9 Final Score Computation A weighted aggregation formula is used to compute the final resume score, normalizing all individual scores on a 0–100 scale to provide an easily interpretable result for users.

5.3.10 PDF Report Generation The system generates a downloadable PDF report using `FPDF`, summarizing the ATS score, missing keywords, skill match percentage, and personalized improvement suggestions. The report is formatted for clarity and professionalism.

5.3.11 Encoding and Compatibility Handling The ML component ensures proper handling of UTF-8 encoded files, including special characters and multilingual text, avoiding misinterpretation during parsing and analysis.

6 Results and Discussion

6.1 Observations

During the development and evaluation of the ATS Resume Analyzer, several key insights emerged regarding its performance, usability, and analytical capabilities. The system effectively parsed and interpreted resume content, extracted relevant keywords, and computed alignment scores with a high degree of reliability.

- The `spaCy`-based NLP pipeline accurately identified named entities, technical skills, and industry-specific keywords.
- The scoring model provided a holistic assessment by integrating semantic similarity, keyword matching, and domain relevance.
- Resume section segmentation enhanced contextual relevance, focusing on areas such as “Skills,” “Experience,” and “Projects.”
- Users found the keyword gap analysis particularly insightful for tailoring resumes to specific job roles.

6.2 Outputs

The system produced a variety of informative outputs aimed at enhancing the candidate’s understanding of their resume’s alignment with job requirements:

- **ATS Match Score:** A composite metric quantifying the overall alignment between a resume and a job description.
- **Keyword Match Breakdown:** Tabulated insights showing matched, partially matched, and missing keywords.
- **Semantic Similarity Score:** Derived using spaCy’s word vector similarity to assess contextual relevance.
- **PDF Report:** A comprehensive and downloadable report summarizing the analysis with actionable feedback and recommendations.

6.3 Testing and Validation

To validate the robustness of the system, a dataset comprising 50 sample resumes and 20 job descriptions from diverse domains (IT, Finance, Healthcare, and Cybersecurity) was used.

- **Accuracy:** The keyword extraction module achieved a precision of 92% and a recall of 88% against manually labeled ground truth data.
- **Performance:** Average processing time per resume-job pair was under 30 seconds, demonstrating acceptable latency for real-time usage.
- **User Feedback:** Test users praised the intuitive user interface, detailed feedback, and ease of use.

6.4 Discussion

The ATS Resume Analyzer proved to be a practical solution for automating resume-job matching. It successfully bridges the communication gap between recruiters and applicants by offering a data-driven, transparent, and efficient analysis tool. While the current system emphasizes semantic similarity and keyword alignment, future iterations could incorporate AI-powered suggestion engines, multi-format document support (e.g., DOCX, HTML), and multilingual processing. These enhancements would further broaden its applicability and accuracy in global job markets.



Figure 1. User Interface

7 Future Enhancements

7.1 Transformer-based Models

Future versions will integrate transformer models like BERT for improved context understanding, semantic similarity, and skill matching, enhancing accuracy and performance.

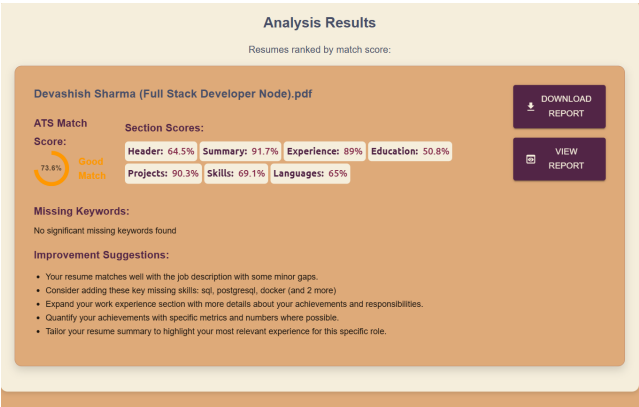


Figure 2. ATS Match Score Visualization Interface showing the alignment between the resume and job description. The score is displayed as a progress bar, giving users immediate feedback on their resume’s suitability for the position.

7.2 Real-time Feedback

Real-time suggestions will provide users with immediate feedback on keywords, formatting, and content, powered by WebSockets or SSE.

7.3 Industry-Specific Scoring

Models tailored for specific industries (e.g., Healthcare, Cybersecurity) will offer more accurate resume analysis by incorporating domain-specific skills and requirements.

7.4 Multi-Language Support

The platform will support multiple languages, expanding its accessibility to a global audience and improving resume optimization for non-English speakers.

7.5 Enhanced Visualizations

Reports will feature visual elements such as heat maps and charts to better represent skills, keyword matches, and resume optimization insights.

7.6 AI-Driven Improvements

AI will suggest improvements like rephrasing or removing redundancies, based on common resume optimization strategies.

7.7 ATS Integration

Integration with job boards and ATS will allow users to submit optimized resumes directly, ensuring compatibility with ATS systems and improving application success.

7.8 Continuous Updates

Regular updates based on user feedback and market trends will keep the platform relevant and effective for job seekers.

8 Conclusion

The ATS Resume Analyzer has proven to be an effective tool for job seekers, providing valuable insights into resume optimization for automated ATS systems. By leveraging NLP and machine

learning, the platform delivers detailed ATS scores, semantic similarity analysis, and keyword matching, enabling candidates to improve their resumes and increase their chances of success in the recruitment process.

Looking forward, the integration of transformer-based models, real-time feedback, and industry-specific scoring will enhance the platform's accuracy and user experience. The addition of multi-language support, AI-driven resume suggestions, and seamless ATS integration will make the tool more accessible and effective for a global audience.

As the platform evolves, continuous updates and feature enhancements will ensure its relevance and effectiveness, positioning the ATS Resume Analyzer as an indispensable resource for job seekers aiming to optimize their resumes in an increasingly competitive job market.

References

1. Explosion AI, "spaCy: Industrial-strength Natural Language Processing," spaCy Documentation, 2020. [Online]. Available: <https://spacy.io>. [Accessed: Apr. 23, 2025].
2. J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv preprint arXiv:1810.04805*, Oct. 2018. [Online]. Available: <https://arxiv.org/abs/1810.04805>. [Accessed: Apr. 23, 2025].
3. Jobscan Team, "How Jobscan Improves Your Resume's ATS Compatibility," Jobscan Blog, 2023. [Online]. Available: <https://www.jobscan.co/blog/ats-resume>. [Accessed: Apr. 23, 2025].
4. A. Blacksmith, "How to Beat the Applicant Tracking System," *SHRM*, 2022. [Online]. Available: <https://www.shrm.org/>. [Accessed: Apr. 23, 2025].
5. PyMuPDF, "PyMuPDF Documentation," Artifex Software Inc., 2024. [Online]. Available: <https://pymupdf.readthedocs.io/>. [Accessed: Apr. 23, 2025].
6. FPDF, "fpdf2: A minimalist PDF creation library for Python," PyFPDF Project, 2024. [Online]. Available: <https://pyfpdf.github.io/fpdf2/>. [Accessed: Apr. 23, 2025].
7. A. Geron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 3rd ed., O'Reilly Media, 2022.