AI-Powered Resume Builder with ATS Optimization Using Natural Language Processing and Generative AI

Nagarajan S[1], Manikandan A[2], Rahul A[3], Nidish M[4], Balaganesan M[5]
[1]*Associate Professor, Department of Computer Science and Engineering*
[2,3,4,5]*Student, Department of Computer Science and Engineering*
Government College of Engineering, Srirangam, Tamil Nadu, India

***Abstract*—In the modern recruitment ecosystem, Applicant Tracking Systems (ATS) play a crucial role in filtering and ranking resumes based on keyword relevance, formatting, and job-specific requirements. However, many candidates fail to pass ATS screening due to lack of awareness about optimization techniques. This paper proposes an AI-powered Resume Builder that integrates ATS score analysis, Natural Language Processing (NLP), and Generative AI to enhance resume quality. The system enables users to create, upload, and optimize resumes while providing real-time ATS scoring and actionable feedback. It also supports automated content generation such as professional summaries and cover letters. Experimental results indicate a significant improvement in ATS scores and resume effectiveness after AI-based enhancement. The system reduces manual effort, improves user experience, and increases the probability of job selection in competitive environments.**

***Index Terms*—Applicant Tracking System, Artificial Intelligence, Natural Language Processing, Resume Optimization, Generative AI.**

I. INTRODUCTION

The recruitment process has undergone significant transformation due to digitalization and automation. Organizations increasingly rely on Applicant Tracking Systems (ATS) to handle large volumes of applications efficiently. These systems evaluate resumes based on keyword matching, formatting, and relevance to job descriptions.

Despite their advantages, ATS systems introduce challenges for job seekers, particularly students and fresh graduates who lack knowledge about resume optimization. Traditional resume-building approaches are manual, time-consuming, and often fail to meet ATS requirements. As a result, many qualified candidates are filtered out before reaching recruiters.

To overcome these limitations, this research introduces an AI-powered resume builder that integrates resume creation, ATS evaluation, and automated content enhancement. The system leverages Natural Language Processing (NLP) techniques to analyze resume content and compare it with job descriptions. Additionally, Generative AI is used to improve writing quality and generate professional content such as summaries and cover letters. The proposed system aims to bridge the gap between candidate skills and ATS requirements by providing intelligent recommendations, thereby improving job selection outcomes.

II. LITERATURE REVIEW

Recent studies have explored the use of Artificial Intelligence in resume analysis and job matching systems. Kokade et al. (2025) proposed an ATS-based resume analyzer using keyword matching techniques, which improved resume evaluation but lacked contextual understanding. Similarly, Jakirhusen et al. (2025) developed a system combining resume analysis and job recommendation using NLP and machine learning, though system complexity increased significantly.

Dubey et al. (2024) introduced an AI-based resume generator that provides real-time suggestions, improving user experience but requiring manual refinement. Earlier work by Mareddy et al. (2021) utilized TF-IDF and cosine similarity for resume ranking, which improved screening efficiency but relied heavily on keyword matching. Although existing systems provide partial solutions, they lack integration of ATS scoring, AI content generation, and real-time feedback in a unified platform.

Recent advancements in Natural Language Processing and deep learning have significantly improved the performance of resume analysis systems. Transformer-based models such as BERT and GPT have enabled better contextual understanding of textual data, allowing systems to move beyond simple keyword matching and perform semantic analysis. These models can identify relevant skills and experiences even when different terminologies are used, thereby improving the accuracy of resume evaluation. The proposed system builds upon these advancements by combining NLP techniques with Generative AI to provide both analytical and generative capabilities in a unified platform.

III. METHODOLOGY

A. System Architecture

The proposed system follows a three-tier architecture consisting of the presentation layer, application layer, and data layer to ensure scalability and efficient processing. The presentation layer, developed using React.js, provides an interactive interface for users to create, edit, and preview resumes. The application layer, built with Node.js and Express.js, handles business logic, processes user requests, and communicates with external AI services for ATS scoring and content generation. The data layer uses MongoDB to store user information, resumes, and generated content in a structured format. Additionally, the system integrates AI APIs to perform resume analysis, keyword extraction, and automated content enhancement, enabling intelligent and real-time feedback for users.

Each layer operates independently, allowing easy maintenance and future upgrades without affecting the overall system functionality. The backend communicates with AI services through well-defined APIs, ensuring seamless processing of resume data and generation of accurate results. Additionally, the system is capable of handling multiple user requests simultaneously, making it suitable for real-world deployment.

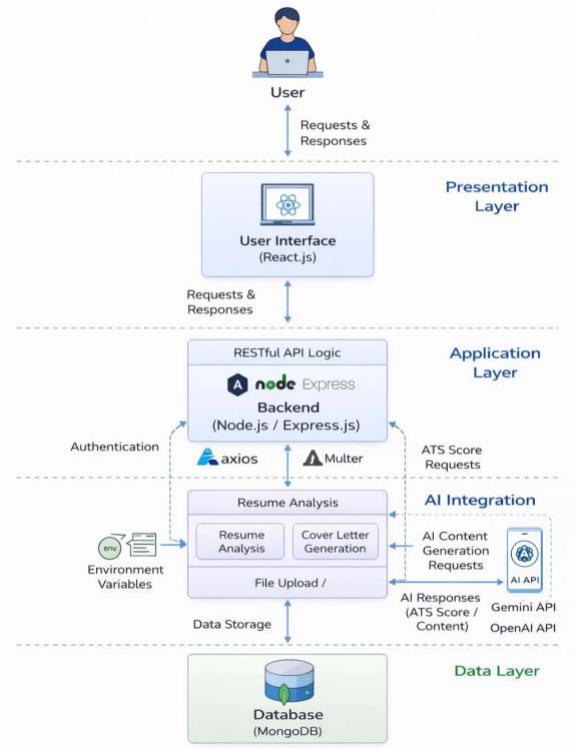


Fig. 1: System Architecture

B. ATS Scoring Mechanism

The ATS scoring mechanism is designed to evaluate the effectiveness of a resume based on its relevance to a given job description. The system calculates the score by analyzing keyword matching, content quality, and structural completeness of the resume. The primary factor considered is the presence of relevant keywords that align with the job requirements.

$$ATS\ Score = (Matched\ Keywords / Total\ Keywords) \times 100$$

In addition to keyword matching, the system also considers other important factors such as proper formatting, inclusion of essential sections (education, experience, skills), and the use of industry-specific terminology. The scoring mechanism uses NLP techniques to identify synonyms and semantically similar terms, ensuring that the evaluation is not limited to exact keyword matches.

C. AI-Based Resume Processing

The AI-based resume processing module leverages Generative AI models to enhance and automate various aspects of resume creation and optimization. The system intelligently improves professional summaries by making them more impactful, concise, and aligned with industry standards. It also refines job descriptions by incorporating strong action verbs, measurable achievements, and ATS-friendly keywords. In addition, the system generates personalized and professional cover letters based on job descriptions and user details, reducing manual effort. Furthermore, it extracts structured information from uploaded resumes, converting unstructured text into organized data such as skills, experience, education, and personal details.

D. Workflow

- User inputs or uploads resume
- Resume data is processed and structured
- NLP extracts keywords and features
- AI analyzes resume and calculates ATS score
- Suggestions and improvements are generated
- Results are displayed to the user

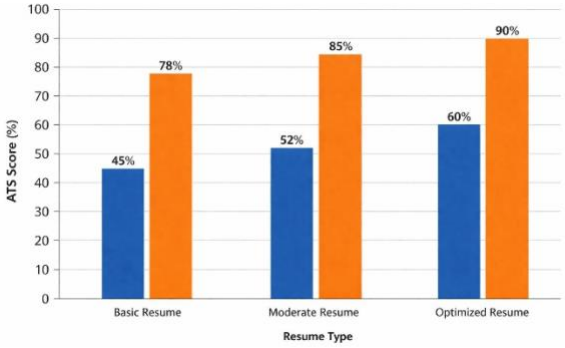


Fig. 2: Flow Diagram

IV. RESULTS AND DISCUSSION

The system was evaluated using different resume samples to measure the improvement in ATS scores before and after AI-based optimization. The results show a significant increase in ATS scores for all types of resumes.

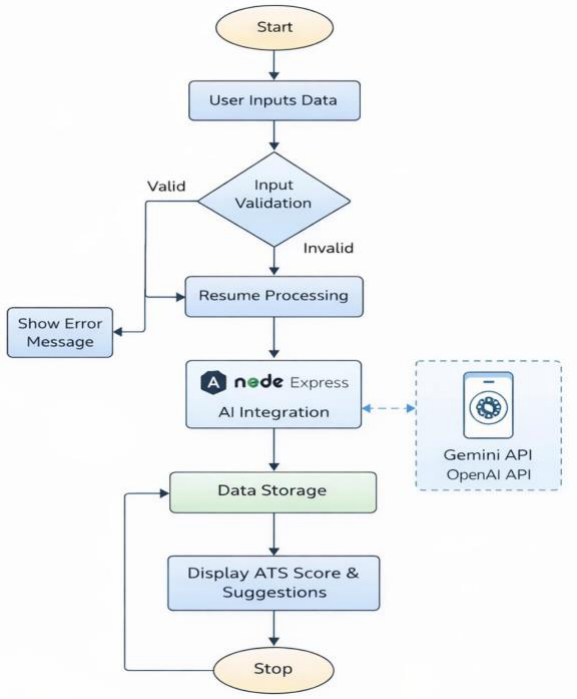


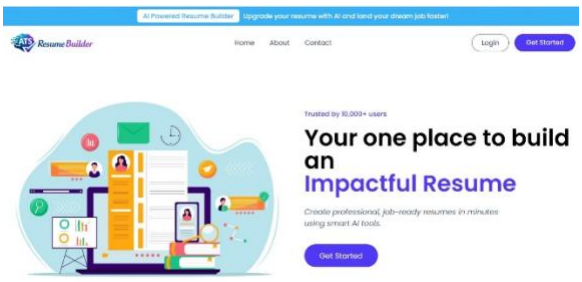
Fig. 3: ATS Score Optimization Result

Key observations from the evaluation include: AI significantly improves keyword relevance; resume quality increases with automated suggestions; and users achieve higher ATS compatibility. The results demonstrate that the proposed system effectively enhances resume performance and increases selection probability.

V. IMPLEMENTATION

A. Frontend Implementation

The frontend of the system is developed using React.js, which provides a dynamic and interactive user interface for resume creation and management.



The application is structured using reusable components and form-based inputs that allow users to enter personal details, education, skills, projects, and work experience efficiently. It also includes features such as real-time preview, template selection, and easy navigation between different sections, enabling users to visualize their resumes instantly while editing.

Fig. 4: Frontend Implementation – Home Page

B. Backend Implementation

The backend is implemented using Node.js and Express.js, which handle all server-side logic and operations. It provides RESTful APIs for user authentication, resume management, and data processing. The backend processes user requests, communicates with the database, and interacts with AI services for ATS analysis and content generation. Secure authentication is ensured using JSON Web Tokens (JWT), allowing only authorized users to access their data.

C. Database Implementation

MongoDB is used as the database to store user information, resumes, and generated content. It is a NoSQL database that supports flexible and scalable data storage, making it suitable for handling dynamic resume structures. The system uses Mongoose as an Object Data Modeling (ODM) tool to define schemas and perform database operations such as insertion, updating, and retrieval efficiently.

D. AI Integration Implementation

Artificial Intelligence is integrated into the system using external AI APIs to enhance resume quality and automate content generation. The system analyzes resumes based on keyword relevance, formatting, and job description matching to generate ATS scores. It also provides suggestions for improvement and generates professional summaries and cover letters. This module plays a key role in improving the effectiveness of resumes.

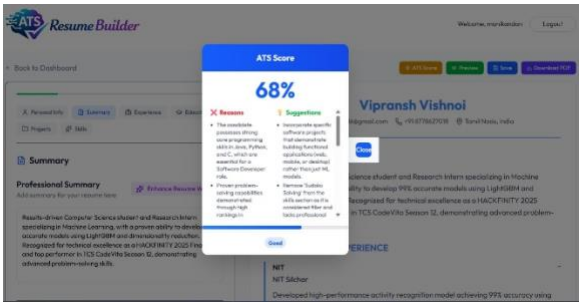


Fig. 5: AI Integration – ATS Score Page

E. File Upload and Processing

The system supports resume upload functionality, allowing users to upload PDF or DOCX files. The uploaded files are processed using text extraction

techniques, and the extracted content is analyzed using AI. The system converts unstructured resume data into structured format, enabling further analysis and optimization without manual data entry.

Fig. 6: File Upload Page

F. Cover Letter Generation Module

The system includes a cover letter generation module that uses AI to create professional and personalized cover letters based on job descriptions and user inputs. This feature helps users save time and ensures high-quality, well-structured content aligned with industry standards.

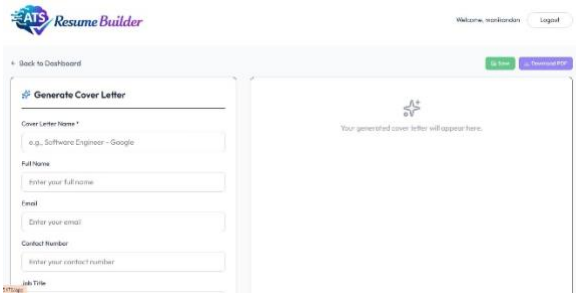


Fig. 7: Cover Letter Page

G. System Integration

All modules of the system, including frontend, backend, database, and AI services, are integrated to ensure smooth communication and efficient data flow. The system supports real-time updates, seamless user interaction, and accurate processing of resume data, providing a complete and user-friendly experience.

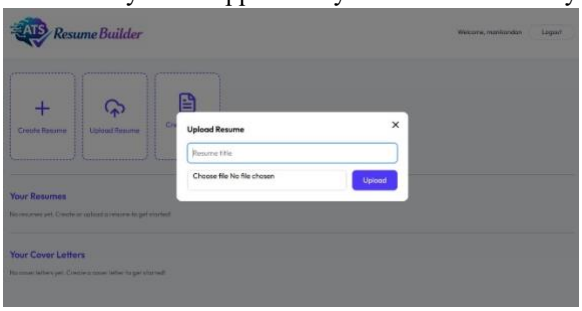
VI. CONCLUSION

The proposed AI-powered Resume Builder with ATS optimization provides an effective solution to the challenges faced by job seekers in modern recruitment processes. By integrating Natural Language Processing (NLP) and Generative AI, the system is capable of analyzing resumes, identifying key areas for improvement, and enhancing content to meet ATS requirements. The inclusion of features such as real-time preview, automated content generation, and ATS score analysis significantly improves the quality and effectiveness of resumes.

The experimental results demonstrate that resumes processed through the system achieve higher ATS scores compared to manually created resumes, thereby increasing the chances of candidate selection. Additionally, the system reduces the time and effort required for resume preparation while ensuring professional and structured output. Overall, the proposed solution successfully bridges the gap between user skills and recruiter expectations, making it a valuable tool for students and professionals in a competitive job market.

VII. FUTURE SCOPES

The proposed system can be further enhanced by integrating advanced features to improve its functionality and applicability. Future work may



include integration with online job portals to enable direct job application and profile matching based on user resumes. Additionally, the implementation of advanced deep learning models can improve semantic understanding and contextual analysis, leading to more accurate ATS scoring and recommendations. The system can also be extended by developing a mobile application to enhance accessibility and user convenience. Furthermore, support for multiple languages can be incorporated to make the system usable for a wider global audience. Another potential enhancement is the inclusion of real-time job recommendations based on user profiles and resume content, which would provide a more personalized experience for users.

REFERENCES

[1] S. Kokade and Y. Nikam, “Resume analyzer with ATS score using AI,” International Journal of Engineering Research, 2025.

[2] S. T. Jakirhusen, R. Shaikh, and M. Patel, “AI-powered resume builder and job recommender system,” International Journal of Computer Applications, 2025.

[3] P. Dubey, A. Sharma, and R. Singh, “AI-based resume generator with intelligent suggestions,” International Journal of Advanced Research in Computer Science, 2024.

[4] S. Mareddy, K. Reddy, and V. Kumar, “Smart resume analyzer using NLP techniques,” IEEE Conference on Data Science, 2021.

[5] T. Mikolov et al., “Efficient estimation of word representations in vector space,” arXiv preprint arXiv:1301.3781, 2013.

[6] J. Devlin, M. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” NAACL, 2019.

[7] A. Radford et al., “Language models are few-shot learners,” Advances in Neural Information Processing Systems (NeurIPS), 2020.

[8] OpenAI, "GPT-4 Technical Report," arXiv preprint arXiv:2303.08774, 2023.

[9] OpenAI, "ChatGPT: Optimizing language models for dialogue," 2022. [Online]. Available: https://openai.com

[10] MongoDB Inc., "MongoDB Documentation," [Online]. Available: https://www.mongodb.com

[11] React Documentation, "React – A JavaScript library for building user interfaces," [Online]. Available: https://react.dev

[12] Node.js Foundation, "Node.js Documentation," [Online]. Available: https://nodejs.org

[13] Yang et al., "XLNet: Generalized Autoregressive Pretraining for Language Understanding," NeurIPS, 2020.

[14] K. Clark et al., "ELECTRA: Pre-training Text Encoders as Discriminators," ICLR, 2020.

[15] Y. Liu et al., "Pre-trained Language Models for Resume Screening and Job Matching," IEEE Access, 2021.

[16] M. Chen et al., "Evaluating Large Language Models for Text Generation," arXiv preprint, 2021.

[17] OpenAI, "GPT-3: Language Models and Their Applications," 2021.

[18] Google Research, "Advances in Natural Language Understanding using Transformer Models," 2022.

[19] A. Brown et al., "AI in Recruitment: Resume Screening and Candidate Selection," Springer, 2022.

[20] S. Kumar and R. Patel, "Automated Resume Screening using Machine Learning Techniques," International Journal of AI Research, 2023.