



Resume Intelligence System

Mr.K.Kiran Babu¹, B.Sanjay², Deepika Gudla³, E.Hansika⁴, Y.Dinesh⁵.

¹ Assistant Professor, Department of CSE (Data Science), ACE Engineering College, Hyderabad, Telangana, India

^{2,3,4,5} III B.Tech. Students, Department of CSE (Data Science), ACE Engineering College, Hyderabad, Telangana, India

How to Cite this Article:

B.Sanjay, , Gudla, D., E.Hansika, & Y.Dinesh, (2026). Resume Intelligence System. International Journal of Creative and Open Research in Engineering and Management, <i>02</i>(04).

<https://doi.org/10.55041/ijcope.v2i4.118>

License:

This article is published under the terms of the Creative Commons Attribution 4.0 International License (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

© The Author(s). Published by International Journal of Creative and Open Research in Engineering and Management.



<https://doi.org/10.55041/ijcope.v2i4.118>

Abstract:

The **Resume Intelligence System** is an AI-driven application designed to automate the lifecycle of resume management, including creation, analysis, and evaluation. By leveraging Natural Language Processing (NLP) and Machine Learning (ML) techniques, the system bridges the gap between job seekers and recruitment standards. An AI-Powered Approach to Recruitment and Career Assistance.

The core functionality allows users to generate or upload resumes, from which relevant keywords and professional skills are extracted using text preprocessing and feature extraction methods such as TF-IDF. A key component of the platform is its inbuilt Applicant Tracking System (ATS) module, which calculates a similarity score to evaluate resume compatibility with specific company job descriptions.

Beyond simple matching, the system employs a machine learning model to predict a candidate's company eligibility and identifies specific "skill-gaps," providing actionable suggestions to enhance employ ability. Furthermore, the platform includes an AI-assisted correction feature for real-time resume refinement. By integrating automated ATS analysis, eligibility prediction, and intelligent feedback into a single unified platform, this system provides a comprehensive, data-driven solution for modern recruitment and professional career development.

1. Introduction:



The Resume Intelligence System is an AI-powered application designed to simplify and enhance the resume screening and career development process. In today's competitive job market, recruiters receive thousands of resumes, making manual evaluation time consuming and inefficient.

This system uses Natural Language Processing (NLP) and Machine Learning (ML) techniques to analyze resumes, extract relevant skills, and evaluate their compatibility with job descriptions through an Applicant Tracking System (ATS). It also predicts company eligibility and provides skill-gap recommendations, helping candidates improve their employ-ability while supporting recruiters with faster and smarter decision-making.

2. Related Work:

Several studies have explored the use of Machine Learning and Natural Language Processing (NLP) in resume screening and recruitment automation. Khatri et al. proposed an AI-based resume screening system that utilizes NLP techniques to extract relevant candidate information and match it with job descriptions, improving hiring efficiency [1]. Tayal et al. developed a resume classification model using algorithms such as K-Nearest Neighbors (KNN) and Support Vector Machines (SVM), demonstrating better accuracy compared to traditional screening methods [2]. Jafari introduced an automated resume ranking system using machine learning models like logistic regression and random forest, enabling efficient candidate evaluation [3]. Other research highlights the use of NLP techniques such as Named Entity Recognition (NER) and text preprocessing for extracting structured data from unstructured resumes [4]. Additionally, ATS-based systems have been widely used to calculate compatibility scores between resumes and job descriptions, improving selection accuracy [5]. Recent advancements also include deep learning approaches such as Long Short-Term Memory (LSTM) networks for better contextual understanding of resume data [6]. Comparative studies indicate that AI-driven recruitment systems significantly outperform manual methods in terms of speed, scalability, and accuracy [7]. However, existing systems still face challenges such as dependency on dataset quality, keyword-based limitations, and lack of semantic understanding. These research works highlight the importance of intelligent systems in modern recruitment while identifying areas for further improvement.

2.1 Existing System and its Limitations:

Title	Technology	Limitation	Authors	Year
Explainable AI for Resume Classification	Uses XAI techniques with ML classifiers resume selection.	Added computational overhead and interpretation.	J. Thomas	2024
Automated Resume Screening Using ML	Uses NLP and classification models to filter resumes based on job requirements	Limited explain-ability of rejection decisions.	R. Sharma, P. Das	2023
Automated Candidate Screening Using BERT	Uses BERT to extract contextual meaning from resumes and rank data.	Requires high GPU resources and large training data.	P. Nair	2023
AI-Based Resume Ranking System	Ranks candidate resumes using skill matching and similarity scoring.	Needs large labeled dataset.	K. Verma	2022



Resume Skill Extraction with Named Entity Recognition	Applies NER models to identify skills, education, experience resumes.	Accuracy drops for non-standard resume formats.	D. Chen, A. Kulkarni	2022
Resume parser with Information Extraction	Extracts skills, education, and experience using NLP parsing techniques	Parsing errors for complex resume formats	S. Iyer, M. Khan	2021
Smart Recruitment System Using ML	Predicts candidate suitability using supervised ML algorithms.	Bias possible in training data.	A. Gupta	2020
Intelligent CV analysis for recruitment	Parses CV's using keyword extraction, weight scoring model.	Keyword dependency may miss relevant profiles.	M. Ali, R. Singh	2020

3. Methodology:

The proposed Resume Intelligence System follows a structured methodology to automate resume analysis using Machine Learning and Natural Language Processing techniques. Initially, resumes are collected in formats such as PDF or DOC and processed using a text extraction module to convert unstructured data into readable text. The extracted text is then preprocessed through NLP techniques including tokenization, stop-word removal, and normalization to improve data quality. Feature extraction is performed using the TF-IDF method to identify important keywords and represent the resume in a numerical format. The processed data is then compared with job descriptions using an Applicant Tracking System (ATS) module to calculate a compatibility score based on skill matching. A Machine Learning model, such as an XGBoost classifier, is trained on a labeled dataset to predict suitable job roles for candidates based on extracted features. The system also performs skill gap analysis by comparing candidate skills with required job skills and identifies missing or weak areas. Based on this analysis, personalized recommendations are generated to help users improve their resumes. The backend is implemented using Flask to handle API requests, while the frontend is developed using Flutter to provide a user-friendly interface. The integration of these components ensures real-time processing, efficient data handling, and accurate results. This methodology enables the system to deliver a reliable, scalable, and intelligent solution for resume screening and recruitment.

3.1 Data Collection and Preprocessing:

The system uses a dataset (*AI_Resume_Screening.csv*) containing resumes and corresponding job roles from technical domains like Data Science, Machine Learning, and Software Development. It also accepts real-time user-uploaded resumes in formats such as PDF, DOC, and DOCX for analysis. Text is extracted from resumes using tools like PyPDF2 to convert unstructured data into readable format. The extracted text is converted to lowercase to maintain consistency. Tokenization is applied to split the text into individual words or terms. Stop words such as “the”, “is”, and “and” are removed to eliminate unnecessary data. Punctuation marks and special characters are removed to clean the text. Duplicate and irrelevant information is filtered out to improve data quality. Normalization techniques like stemming or lemmatization are applied to standardize words. Important keywords and skills are identified using NLP techniques. The TF-IDF method is used to convert textual data into numerical feature vectors. The processed data is structured and prepared for Machine Learning model training. This preprocessing step improves accuracy, efficiency, and overall system performance.



3.2 Feature Extraction:

The Resume Intelligence System can be further enhanced with advanced features to improve its performance and usability. Future improvements may include supporting multiple resume formats such as PDF, DOC, and DOCX with higher parsing accuracy. More advanced Machine Learning and Deep Learning models can be integrated to improve job role prediction and ATS scoring. The system can also be connected to real-time job portals to provide live job recommendations based on the candidate's profile. Additional features like personalized resume improvement suggestions can guide users in increasing their ATS scores. The application can be extended into a mobile app for better accessibility and user experience. Furthermore, adding multi-language support will make the system more inclusive and useful for users from different regions. Overall, with continuous improvements, the system has strong potential to become a complete intelligent recruitment assistant.

3.3 Model Selection and Training:

- A suitable Machine Learning model is selected to predict job roles based on resume data.
- The XGBoost Classifier is chosen due to its high accuracy and efficiency in classification tasks. The dataset is divided into training and testing sets using an 80:20 ratio.
- Feature vectors are generated using TF-IDF to represent resume text numerically.
- Label encoding is applied to convert job roles into numerical form.
- The model is trained on the training dataset using extracted features and labels.
- During training, the model learns patterns between skills and corresponding job roles.
- Hyperparameters are tuned to improve model performance and avoid overfitting.
- The trained model is evaluated using the testing dataset.
- Performance metrics such as accuracy are used to measure model effectiveness.
- The model shows high accuracy in predicting technical job roles.
- Once trained, the model is integrated into the system for real-time predictions.
- The model continuously improves with better data and feature selection.

3.4 Feature Engineering and Selection:

- Raw resume text is processed using NLP techniques to extract meaningful information.
- Tokenization is applied to split text into individual words.
- Stop words and unnecessary data are removed to reduce noise.
- Important keywords and skills are identified from the resume content.
- TF-IDF is used to convert text into numerical feature vectors.
- Relevant features are selected while removing less important ones to reduce dimensionality.
- The final optimized feature set improves model accuracy and performance.

3.5 Model Evaluation:

The trained model is tested using a separate portion of the dataset to check how well it works on new data. Accuracy is used to understand how correctly the model predicts job roles from resumes. The results show that the model performs well and gives correct predictions in most cases. A confusion matrix is used to see where the model is making correct or wrong predictions. Other measures like precision and recall help in understanding the quality of the predictions more clearly. The model works especially well for technical job roles, but its performance can slightly change depending on the quality of the data used. Overall, the evaluation shows that the model is reliable and suitable for automating the resume screening process.



3.6 Comparison with Baseline Methods:

The proposed Resume Intelligence System is compared with basic or traditional methods such as manual resume screening and simple keyword-based filtering. In manual screening, recruiters spend a lot of time reviewing resumes, which can lead to errors and inconsistency. Keyword-based methods, on the other hand, only check for the presence of certain words and may miss candidates with relevant skills described differently. In contrast, the proposed system uses Machine Learning and NLP techniques to understand resume content more effectively. It not only identifies important skills but also predicts suitable job roles and provides ATS scores. The system performs faster and gives more consistent results compared to baseline methods. It also reduces human effort and improves accuracy in candidate selection. Overall, the proposed approach shows better performance, efficiency, and reliability than traditional resume screening methods.

3.7 Ethical Considerations:

The Resume Intelligence System is designed with important ethical considerations in mind to ensure fairness and responsible use. One key concern is avoiding bias in the model, as the system's predictions depend on the data it is trained on. If the dataset is biased, it may lead to unfair results, so care is taken to use balanced and diverse data. The system also protects user privacy by securely handling resumes and personal information. It does not misuse or share sensitive data without permission. Transparency is another important factor, where users should have a basic understanding of how their resumes are being evaluated. Additionally, the system is designed to assist recruiters and candidates, not replace human decision-making completely. Human involvement is still important to ensure fair and final hiring decisions. Overall, the system aims to provide a fair, secure, and responsible solution for recruitment.

3.8 Result:

The Resume Intelligence System show that it works effectively in automating resume analysis and predicting job roles. The system is able to process resumes by extracting text and identifying important technical skills using NLP techniques. By using TF-IDF, it improves how accurately keywords are detected, which helps in better analysis. The Machine Learning model, trained with an 80:20 data split, performs well and gives accurate job role predictions. It also calculates ATS scores by matching skills with job descriptions, making it easier to understand how suitable a resume is. The system works particularly well for technical roles like Data Analyst, Software Developer, and Machine Learning Engineer. The integration of the Flask backend with the Flutter frontend ensures smooth operation and quick results. It processes resumes in a short time, making the system efficient and scalable. The skill extraction module helps in identifying relevant skills while reducing unnecessary information. It also suggests suitable companies based on the candidate's profile, which adds more value. However, the performance depends on the quality of the dataset and predefined skills. It may not work as effectively for non-technical roles or complex resume formats. Even with these limitations, the system proves to be reliable and useful. Overall, it successfully improves the recruitment process through intelligent automation.



Resume Intelligence System

Create Resume

Upload Resume & Analyze



Select Resume Template



Professional Resume Template

Use Template



← Contact Information

Name

Address

Email

Mobile Number

SAVE & NEXT

← Academic Details

Course / Degree

School / College / University

Marks %

Year of Passing



SAVE & NEXT



← Work Experience

Job Title

Company Name

Start Year

End Year

Description



SAVE & NEXT

← Project Details

Project Title

Description

Duration

Role

Team Size



SAVE & NEXT



← Skills & Interests

Skills

Skill

+

Interests

Interest

+

SAVE & NEXT

← Achievements & Declaration

Achievements

Achievement

+

Career Objective

Enter your career objective

Declaration

Enter declaration

Place

Date

GENERATE RESUME



← Resume Preview

Career Objective

Skills

Achievements

Declaration

Place: _____ Date: _____

Signature _____

DOWNLOAD PDF

localhost:58351

← Resum

Career Objec

Skills

Achievement

Declaration

Place: _____ Date: _____

Signature _____

Print 1 sheet of paper

Destination: Microsoft Print to PDF

Pages: All

Color: Color

Paper size: Letter

Pages per sheet: 1

Scale: Default

Print using system dialog... (Ctrl+Shift+P)

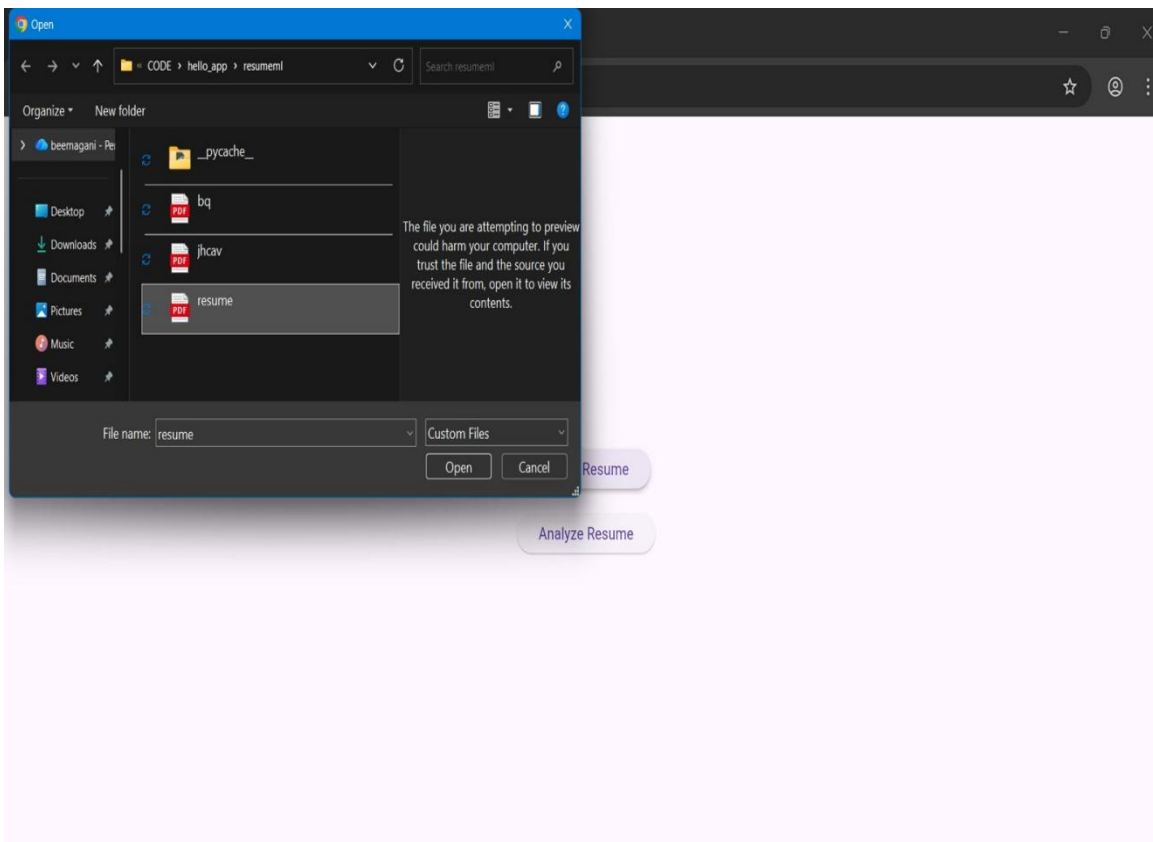
Print Cancel



← Upload & Analyze Resume

Select Resume

Analyze Resume





← Upload & Analyze Resume

Select Resume

Analyze Resume

ATS Score: 40

Predicted Role: Machine Learning Engineer

Recommended Companies

Wipro

TCS

IBM

Conclusion:

Conclusion:

The Resume Intelligence System provides an efficient solution for automating resume analysis using Machine Learning and Natural Language Processing (NLP) techniques. It extracts key information such as skills, qualifications, and experience from resumes and evaluates them by calculating ATS scores based on job-specific keywords. The system helps candidates understand their strengths and areas for improvement while also predicting suitable job roles. It reduces manual effort in recruitment and improves the accuracy and consistency of resume screening. Additionally, it assists recruiters in selecting the most suitable candidates quickly and supports better decision-making. By integrating data processing and predictive analysis, the system delivers reliable results and enhances transparency in the hiring process. Overall, the project demonstrates the practical application of AI in recruitment and serves as a smart and effective tool for modern hiring needs.

References:

Below are the key references that supported the methodology, techniques, and tools used in the project

- 1) Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. Conference: NAACL-HLT. DOI: 10.18653/v1/N19-142. link: [-A Review of Natural-Language-Instructed Robot Execution Systems](#)
- 2) Ramos, J. (2003). *Using TF-IDF to Determine Word Relevance in Document Classification*. Conference: Proceedings of the First Instructional Conference on Machine Learning. Link :- https://www.researchgate.net/publication/228737527_Using_TF-IDF_to_Determine_Word_Relevance_in_Document_Classification



- 3) Pedregosa, F., et al. (2011). *Scikit-learn: Machine Learning in Python*. Journal: Journal of Machine Learning Research (JMLR). Link :- [\[1201.0490\] Scikit-learn: Machine Learning in Python](#)
- 4) Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python*. Publisher: O'Reilly Media. Link :-
- 5) Loper, E., & Bird, S. (2002). *NLTK: The Natural Language Toolkit*. Conference: ACL Workshop
- 6) Reimers, N., & Gurevych, I. (2019). *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*. Conference: EMNLP
- 7) Reimers, N., & Gurevych, I. (2019). *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*. Conference: EMNLP
- 8) McKinney, W. (2010). *Data Structures for Statistical Computing in Python*. Conference: Python in Science Conference
- 9) Grinberg, M. (2018). *Flask Web Development: Developing Web Applications with Python*. Publisher: O'Reilly Media
- 10) Google Developers. (2023). *Flutter Documentation*. Available: <https://docs.flutter.dev>