A Smart Framework for Skill Gap Analysis and Personalized Career Recommendation

TEJASHWINI SHETTIGARI [1], BUSA SANJAY [2], TANKARI SAHITH REDDY [3], S SOWJANYA [4]
[1,2,3,4]*Department of Computer Science and EngineeringVardhaman college of Engineering(Autonomous)Hyderabad,Telangana, India ORCID ID : 0009-0003-2057-2751*

Abstract- The gap between academic education and industry expectations has become a significant challenge for students and job seekers. While employers often look for candidates with specific technical skills, many individuals are unsure whether their resumes properly represent the competencies required for particular job roles. Most existing resume screening tools mainly depend on keyword matching and do not provide meaningful insights about missing skills or areas that need improvement.This paper proposes a Smart Framework for Skill Gap Analysis and Personalized Career Recommendation that evaluates how well a candidate's resume matches the requirements of a selected job role. The system extracts technical skills from an uploaded resume and compares them with predefined industry skill requirements. Based on this comparison, it calculates a match percentage and identifies the skills that are missing. In addition, the framework suggests alternative career paths that align with the candidate's current skill set.The proposed system presents the analysis results in a clear and easy-to-understand format, helping users recognize their strengths and understand which skills need further development. By combining structured data comparison with intelligent processing, this framework aims to improve modern career guidance systems and assist individuals in making more informed career decisions.

Keywords— Skill Gap Analysis, Resume Analysis, Career Recommendation, Skill Matching, Employability Support

I. INTRODUCTION

The job market today is very competitive and continues to change rapidly. New technologies are introduced frequently, and companies often update the skills they expect from potential employees.

Although students learn important theoretical concepts during their academic studies, they may not always know the exact practical skills that industries require. As a result, a gap often forms between the knowledge students gain in college and the skills employers expect from job candidates.

Candidates apply to jobs, sometimes even when they have no idea if their resume fits at all. When they are turned away, they most of the time don't get helpful insights into what went amiss.

Most existing systems of resume screening utilize filtering of resumes by key dans. While these systems help recruiters with sifting through applications, they seldom offer candidates direction on what can be improved.

In order to overcome this issue, this project provides Smart Framework for Skill Gap Analysis and Personalized Career Recommendation. Users can upload their resume and pick a target job role on the system.

It then parses the resume and compares extracted skills to expected industry skills, returning a match percentage. It also focuses on the skills that are missing and suggests alternative job roles that align better with users' existing skill set.

The main goal of this project is to help users clearly understand their strengths, weaknesses, and possible career opportunities using a structured and data-driven approach.

II. RELATED WORK

Several such digital tools exist for the purpose of career guidance, resume analysis, and matching. Professional networking sites use profile analysis for the purpose of recommending jobs, whereas other tools may employ the use of score-based mechanisms for the purpose of analyzing resumes. However, these tools do not provide the analysis of skill gaps.

Studies conducted in the domain of recommendation systems have proved the efficiency of collaborative filtering and content-based filtering for the purpose of matching users. However, the interpretability of such methods is limited.Users can be provided with job recommendations without being aware of the rationale behind the recommendation.

Research works on employability analytics highlight the significance of structured skill mapping and competence models. Machine learning models were also explored by some researchers to predict career paths using historical data.

This approach may demand large datasets and sophisticated training processes, making it difficult to implement at a smaller scale or in an academic context. A second constraint identified with existing systems is that they do not provide adequate feedback on the absence of skills.

Most systems provide binary results like 'matched' or 'not matched' without explanations. This proposed framework is different in that it focuses on clarity and transparency rather than accuracy.

III.PROPOSED METHOD

The proposed framework is designed in a way that the components of the system are modular. Each component of the system is designed to carry out specific but interlinked functionalities.

The first component of the system is the submission of the resume along with the chosen role. This is to ensure that the user is actively engaged in the determination of their career goals.

The text is extracted from the resume through the use of the PDF parsing mechanism. The extracted text is then processed to standardize the case of the letters. This is done to eliminate unnecessary formatting tags.

The extracted skills from the resume are carried out by comparing the extracted text from the resume against the predefined list of technical skills. The use of sets in the system is to eliminate the occurrence of duplicate skills. The extracted skills from the resume are used to retrieve the required skill list for the chosen role.

The compatibility percentage is calculated through the use of a mathematical formula.
The recommendation engine is designed to extend the functionality of the system. The extracted skills from the resume are compared against all the roles in the dataset. The compatibility percentage is calculated for all the roles in the dataset. The roles are then ranked in descending order. The roles at the top of the rank are used as recommendations for the user.

The modular design of the system ensures that any component of the system can be easily modified. For example, the skill extraction component of the system can easily adopt the use of NLP techniques in the future.

IV. IMPLEMENTATION

A. A. System Architecture and Data Sources The proposed Skill Gap Analysis and Career Recommendation System is based on a modular web-based system, where different functional components are independent but communicate with each other smoothly through backend logic.

The system is based on a layered structure, which includes the User Interface layer, Resume Processing, Skill Extraction Engine, Matching Engine, Dataset Handler, and Recommendation Engine. The frontend interface is responsible for uploading the resume and selecting the required job roles.

The backend server, which is based on the proposed system, is implemented using the Python programming language.

Flask is responsible for handling requests and executing operations like skill extraction and comparison. This is a modular design, which allows for scalability and easier debugging in case of unforeseen errors.

© JUL 2026 | IRE Journals | Volume 10 Issue 1 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I1-1719626

The system uses a structured data set with job roles and the required technical skills for each role. This data set is stored in CSV format and handled using the Pandas library. Unlike other systems that rely on real-time data through APIs, the proposed framework mainly depends on static data to provide stability and consistency.

Table 1: Dataset Used

Source	Type	Content	Update Frequency
Job Skills Dataset	CSV File	Job Roles & Required Skills	Manual Update
Resume File	PDF	User Skills	On Upload

B. Resume Processing and Skill Extraction

The system accepts a resume in PDF format as an input. After accepting the resume, the backend processes the text using a PDF library. This text is then preprocessed by changing the case to lowercase and removing unwanted characters.

To extract the skills from the resume, a predefined list of technical skills is used. A keyword-based approach is used to find the skills in the resume text by checking if the skill is present in the text or not. All the skills are stored in a set to avoid duplication.

The extracted skills are stored as the user's skill profile, which is then employed to further compare the skills with the job role requirements:

- Consistency in the identification of skills
- Reduced redundancy
- Clear mapping of resume skills and dataset skills.

C. Skill Matching and Compatibility Quantification

Clear mapping of skills from the resume to the skills present in the dataset.

Let:

- S_r= Resume Skills
- S_j= Job Role Required Skills

Matched skills are calculated as:

$$S_{matched} = S_r \cap S_j$$

Missing skills are calculated as:

$$S_{missing} = S_j - S_r$$

The compatibility percentage is computed as:

Match Percentage=|Smatched|×100

―――――――――

|Sj|

Equation 1: Skill Match Percentage

$$P = \frac{M}{R} \times 100$$

Where:

- M= Number of matched skills
- R= Total required skills

If no skills match, the system assigns:

P=0

This ensures computational safety and transparency. The compatibility score provides a quantitative measure of how well the resume aligns with the selected role.

D. Career Recommendation Engine

Beyond analyzing a single job role, the system evaluates the user's extracted skills against all available job roles in the dataset.

For each role i:

$$P_i = \frac{\mid S_r \cap S_{j_i} \mid}{\mid S_{j_i} \mid} \times 100$$

The recommendation logic remains transparent and interpretable, unlike black-box machine learning systems.

Equation 2: Career Recommendation Score

All job roles are ranked in descending order based on P_i, and the top five are recommended to the user.

This approach ensures:

- Personalized suggestions
- Broader career exploration
- Data-driven role ranking

The recommendation logic remains transparent and interpretable, unlike black-box machine learning systems.

E. Result Interpretation and Feedback Mechanism

The system categorizes compatibility scores into three interpretive levels:

Table 2: Compatibility Interpretation Scale

Match Percentage	Interpretation
>=80%	Highly Suitable
50%-79%	Moderately Suitable
<50%	Needs Improvement

F. End-to-End Execution Pipeline

The complete workflow of the system is as follows:

1. User uploads resume and select job role.
2. Resume text is extracted from the PDF.
3. Technical skills are identified from resume content.
4. Required skills for selected job role are retrieved.
5. Matched and missing skills are computed.
6. Alternative career recommendations are generated.
7. Results are displayed in structured format.

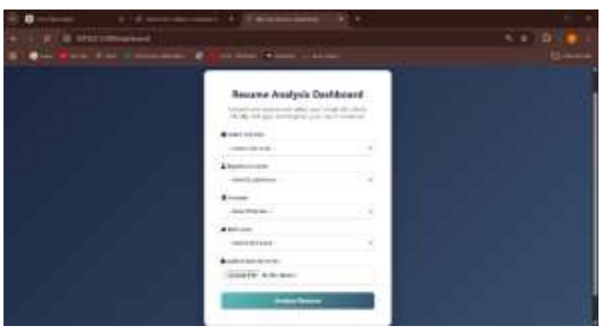

Fig. 1: Sample Resume Input

Fig. 2: Corresponding Output

The system ensures that each step is deterministic and reproducible.

V.EXECUTION

The next sections explain the runtime behavior of the Skill Gap Analysis and Career Recommendation System. It explains the processing of the input, the interaction of the internal modules, and the production of the structured output. Although the previous sections explained the design and the architecture of the system, this section explains the runtime behavior of the system.

A. Runtime Execution Flow

At runtime, the system works as a structured pipeline of executions, where the Flask backend server coordinates the entire process. This commences with the user uploading a resume in the form of a PDF file, along with the desired job role. Other optional information may include the level of experience, domain, and skill level.

Once the user submits the form, the backend checks the uploaded file. It then stores the file temporarily in the upload directory. Next, the resume processing module parses the uploaded file using a PDF parsing engine. The extracted information is then converted to lowercase for standardization purposes.

At the same time, the desired job role is obtained from the job role dataset. Next, the skill extraction engine parses the uploaded resume to identify technical skills present in the dataset. The skills are then stored as a structured data type to prevent redundancy.

After the skills are extracted, the matching engine compares the skills present in the resume with the skills required for the selected job role. The skills that match, as well as the skills that are required but missing, are determined by using set-based operations. Then, the percentage of the compatibility is determined by using the predefined mathematical formula.

Finally, after the role-based analysis is done, the recommendation engine checks the user's skill profile for all the job roles. Then, the percentage of the compatibility is determined for each of the roles, ranked, and the top five recommendations are produced.

B. Service Orchestration and Determinism
The system is designed with modular independence between functional components. Resume parsing, skill extraction, matching logic, and recommendation generation operate as logically separate modules within the backend.

Because the dataset remains static during execution and the algorithm is deterministic, the system produces consistent results when identical inputs are provided. If the same resume and job role are used multiple times, the match percentage and skill breakdown remain unchanged.

Unlike probabilistic AI systems, the framework prioritizes interpretability over randomness or model uncertainty.

C. Execution Assumptions and Constraints
The system operates under specific assumptions to maintain clarity and safety:
1. Skill extraction is keyword-based and assumes that technical skills explicitly appear in the resume text.
2. The job-role dataset defines industry requirements and serves as the primary comparison benchmark.
3. Compatibility percentage reflects skill coverage only and does not measure expertise depth.
4. The system does not perform resume ranking for recruitment purposes but instead provides self-assessment insights.

Additionally, since the dataset is manually curated, periodic updates are required to maintain alignment with evolving industry standards.
These constraints ensure that the system remains a career guidance tool rather than an automated recruitment decision system.

VI.RESULTS AND DISCUSSIONS

A. System Output Characteristics
The system produces structured and interpretable output that includes:
- Match Percentage
- Matched Skills Table
- Missing Skills Table
- Career Recommendations

The compatibility score visually represents the level of alignment between the resume and the selected job role. The matched skills table confirms the strengths already possessed by the user, while the missing skills table highlights areas requiring improvement.

Repeated execution using identical input produces consistent outputs, confirming the stability of the execution pipeline. This consistency strengthens trust in system reliability.

Unlike black-box recommendation systems, the framework emphasizes transparency by explicitly showing how results are computed.

B. Sensitivity to Job Role Selection
To evaluate role-based sensitivity, the same resume was tested across different job roles.

Table 3: Same Resume - Different Job Roles

Resume Skills	Selected Role	Match %	Interpretation
Python, SQL, Pandas	Data Scientist	75%	Moderate
Python, SQL, Pandas	Data Analyst	85%	High
Python, SQL, Pandas	Web Developer	20%	Low

C. Sensitivity to Resume Variation
To test sensitivity to skill differences, different resumes were evaluated for the same job role.

Table 4: Different Resumes- Same Job Role

Resume Profile	Role	Match %
Python, ML, Statistics	Data Scientist	80%
HTML, CSS, JavaScript	Data Scientist	15%

DOI: https://doi.org/10.64388/IREV10I1-1719626

D. Explainability and Skill Decomposition

One of the strongest features of the system is its ability to decompose compatibility into understandable components.

Instead of providing a single opaque score, the system:

- Displays exactly which skills match
- Clearly lists missing competencies
- Quantifies alignment mathematically

This interpretability enhances user confidence and allows individuals to take concrete steps toward improvement.

E. Discussion

The results demonstrate that the proposed framework effectively bridges the gap between resume content and industry-defined job requirements. The system provides structured guidance without replacing human decision-making.

By combining deterministic skill comparison with ranked recommendations, the framework supports both targeted improvement and broader career exploration.

Overall, the findings suggest that the system successfully fulfills its objective of delivering transparent, interpretable, and actionable career insights.

VII.LIMITATIONS AND CHALLENGES

Despite its effectiveness, the system has certain limitations:

1. Skill detection relies on keyword matching and may not capture semantic variations.
2. The dataset may not include emerging technologies or hybrid roles.
3. Soft skills are not deeply analyzed.
4. Expertise depth is not evaluated—only skill presence is considered.
5. Resume formatting complexity may affect text extraction quality.

While these limitations are inherent to the current implementation, they provide opportunities for future enhancement.

The design prioritizes safety and interpretability over predictive automation, ensuring ethical use in career guidance contexts.

VIII.CONCLUSION AND FUTURE SCOPE

This paper presented a Smart Framework for Skill Gap Analysis and Personalized Career Recommendation designed to assist students and job seekers in evaluating their readiness for specific job roles.

The system integrates resume parsing, structured skill datasets, compatibility calculation, and career ranking mechanisms to generate clear and actionable feedback. Its modular architecture ensures stability, reproducibility, and scalability.

A key design principle is the separation between skill compatibility analysis and professional hiring decisions. The system functions as a guidance tool rather than a recruitment filter.

Future Scope

Future improvements may include:

- Integration of advanced NLP for semantic skill recognition
- Incorporation of real-time job market data
- Machine learning–based compatibility prediction
- Cloud-based deployment
- Soft skill evaluation modules
- Personalized learning pathway recommendations

These enhancements could transform the system into a comprehensive intelligent career advisory platform.

REFERNECES

[1] C. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval. Cambridge, U.K.: Cambridge University Press, 2008.

[2] D. Jurafsky and J. H. Martin, Speech and Language Processing, 2nd ed. Upper Saddle River, NJ, USA: Pearson, 2009.

[3] S. Bird, E. Klein, and E. Loper, Natural Language Processing with Python. Sebastopol, CA, USA: O'Reilly Media, 2009.

[4] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," IEEE Transactions on Knowledge and Data Engineering, vol. 17, no. 6, pp. 734–749, Jun. 2005.
[5] F. Ricci, L. Rokach, and B. Shapira, Recommender Systems Handbook, 2nd ed. Boston, MA, USA: Springer, 2015.
[6] M. Paparrizos, B. B. Cambazoglu, and A. Gionis, "Machine learned job recommendation," in Proceedings of the 5th ACM Conference on Recommender Systems (RecSys), Chicago, IL, USA, 2011, pp. 325–328.
[7] A. Singh and S. Jain, "Resume parser using NLP and machine learning techniques," in Proceedings of the International Conference on Smart Systems and Inventive Technology, 2020.
[8] S. K. Pal and S. Mitra, "Career guidance and recommendation system using data mining techniques," International Journal of Computer Applications, vol. 155, no. 3, pp. 1–6, 2016.
[9] R. S. Baker and K. Yacef, "The state of educational data mining in 2009: A review and future visions," Journal of Educational Data Mining, vol. 1, no. 1, pp. 3–17, 2009.
[10] J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, 3rd ed. Burlington, MA, USA: Morgan Kaufmann, 2011.
[12] P. Brusilovsky and E. Millán, "User models for adaptive hypermedia and adaptive educational systems," in The Adaptive Web. Berlin, Germany: Springer, 2007, pp. 3–53.