An Intelligent Job Recommendation System Integrating Resume Parsing, NLP-based Matching, and Skill Gap Visualization

Md. Mahmudol Hasan
Department of Computer Science
Green University of Bangladesh
222902041@student.green.edu.bd

Rudra Roy
Department of Computer Science
Green University of Bangladesh
222902014@student.green.edu.bd

Khadija Begum
Department of Computer Science
Green University of Bangladesh
222902038@student.green.edu.bd

***Abstract*—This paper presents a modular intelligent job recommendation system combining NLP-based resume parsing, TF-IDF and cosine similarity-based job matching, and interactive visualization of candidate skill gaps. The approach improves personalization and transparency over traditional keyword matching, with experimental validation demonstrating increased relevance and user interpretability. Our system integrates secure user authentication, automated resume parsing, semantic job matching, and skill gap visualization to enhance job discovery and provide actionable insights for skill development.**

I. Introduction

In the current competitive job market, job seekers and employers alike face challenges in discovering optimal matches. Conventional job platforms primarily use keyword-based search engines that suffer from semantic mismatches—candidates and jobs may use different terminology for similar skills or experiences. This leads to limited interpretability of recommendations and suboptimal matches.

Our work addresses these shortcomings by integrating automated resume parsing with machine learning-driven semantic matching and intuitive visualization of missing skills. Unlike existing systems that rely solely on keyword matching or collaborative filtering, our framework leverages natural language processing (NLP) techniques to extract structured candidate profiles and represents both resumes and job descriptions in a vector space using TF-IDF features.

The system workflow enables users to securely log in, upload resumes, receive personalized job recommendations ranked by cosine similarity, and view interactive visualizations highlighting skill gaps relative to job requirements. This combination improves job discovery efficacy and empowers candidates with actionable feedback for professional growth.

II. Background and Related Work

Job recommendation systems have traditionally used keyword-based content filtering or collaborative filtering based on user interaction history [7], [8]. However, keyword-based approaches often fail to capture semantic nuances, while collaborative filtering suffers from the cold start problem and limited explainability.

Recent research has explored deep learning embeddings such as BERT [12] for semantic understanding of resumes and job descriptions [11]. While such models improve accuracy, their complexity and reduced interpretability make them less practical for rapid prototyping and smaller datasets.

Many existing works do not combine deep resume parsing with interpretable visualization of skill gaps. Our approach offers a modular architecture facilitating independent updates to parsing, matching, and visualization components. We use TF-IDF vectorization as an efficient and interpretable baseline, allowing transparent similarity calculations and intuitive explanations for recommendations.

III. Methodology

A. User Interface

We implemented the user interface using Streamlit due to its interactive visualization capabilities and rapid prototyping ease. The UI guides the user through the workflow: secure login, resume upload in PDF format, automatic parsing, job matching, and skill gap visualization. This interactive design enhances user engagement and provides clear actionable insights.

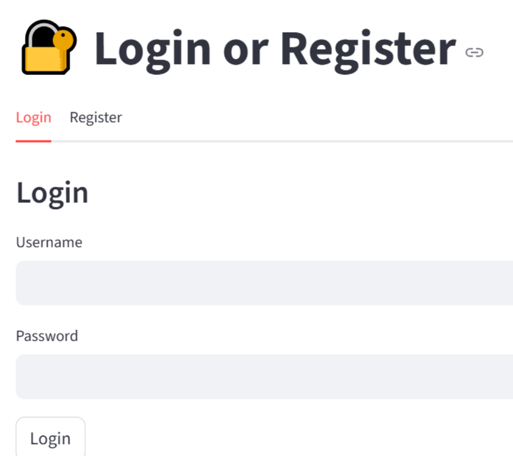

Fig. 1. User Login Interface for Secure Authentication

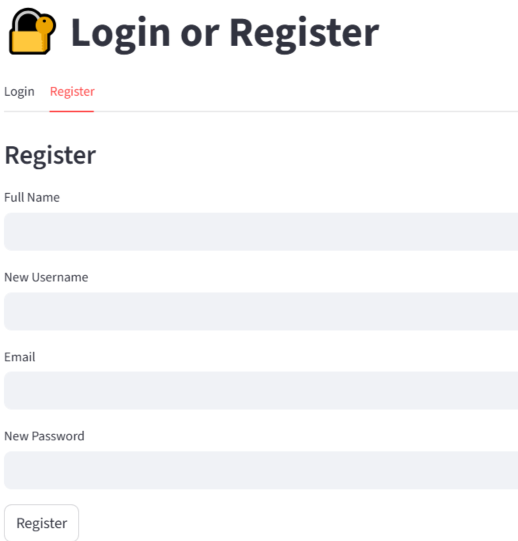

Fig. 2. User Registration Interface.

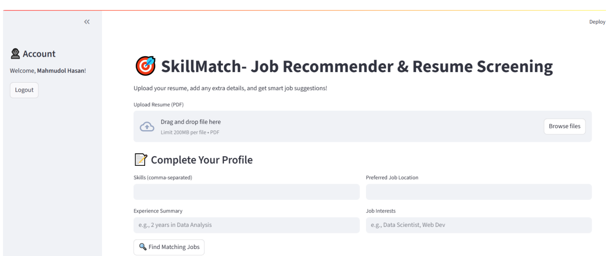

Fig. 3. Dashboard of the Application.

B. Implementation Architecture

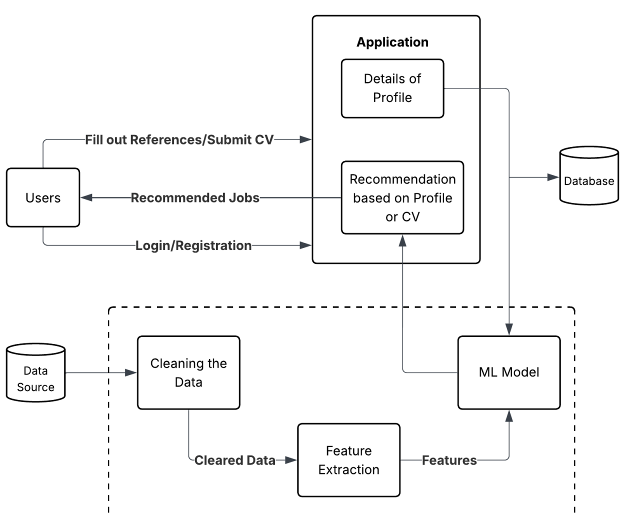

Fig. 4. System Block Diagram.

Our system follows a modular design enabling ease of maintenance and upgrades. Key components include:

- **Authentication Module:** Handles secure user login and registration.
- **Resume Parser:** Extracts structured candidate data from PDF resumes using NLP techniques.
- **Job Dataset:** A curated collection of job postings serving as the recommendation pool.
- **Recommender Engine:** Calculates similarity between resumes and job descriptions for ranking.
- **Visualizer:** Generates interactive charts highlighting skill gaps and match quality.
- **Main Controller:** Coordinates interactions between modules.

This architecture allows independent development and testing of modules, facilitating scalability and adaptability.

C. NLP-based Data Processing

The system preprocesses textual data through:

1) **Text Extraction:** Resume PDFs are converted to raw text using `pdfplumber`.
2) **Tokenization:** Text is segmented into tokens.
3) **Stopword Removal:** Common non-informative words are removed.
4) **Lemmatization:** SpaCy's lemmatizer is applied to reduce words to base forms, improving consistency.
5) **Skill Extraction:** Tokens are matched against a domain-specific skill dictionary to identify candidate skills.
6) **Vectorization:** TF-IDF transforms processed text into numerical feature vectors.

TF-IDF was selected for its interpretability and efficiency on moderate dataset sizes, providing a transparent baseline that can be enhanced with deep embeddings in future work.

D. Machine Learning Model Training

We employ a content-based recommendation approach:

- Resumes and job descriptions are represented in the same TF-IDF feature space.
- Cosine similarity quantifies the semantic proximity between candidate profiles and job postings.
- Jobs are ranked according to similarity scores; top 5 recommendations are presented by default.

Cosine similarity between resume vector R and job vector J is defined as:

$$Sim(R, J) = \frac{R \cdot J}{\|R\| \times \|J\|} \quad (1)$$

This method is computationally efficient and scalable to moderate datasets. For larger datasets, approximate nearest neighbor search could be incorporated to improve performance.

Algorithm 1: Job Matching Using NLP and Cosine Similarity

1) Input: Resume document R, Job descriptions $\{J_1, J_2, ..., J_n\}$.
2) Preprocess texts: tokenize, remove stopwords, lemmatize.
3) Convert R and each J_i into TF-IDF vectors.
4) Compute $Sim(R, J_i)$ for all i.
5) Rank jobs by similarity scores.
6) Output: Top-k job recommendations.

Algorithm 2: Resume Parsing and Skill Extraction

1) Input: Resume PDF document D.
2) Extract raw text T using `pdfplumber`.
3) Apply Named Entity Recognition (NER) via SpaCy to identify candidate name.
4) Extract email and phone number using regular expressions.
5) Match tokens in T against predefined skill set to identify candidate skills.
6) Output: Structured profile {name, email, phone, skills, raw_text}.

Skill Gap Detection is performed by computing the set difference between required job skills and candidate skills, enabling visualization of missing competencies.

IV. Observations and Results

We evaluated the system on a sample dataset of resumes and job postings. The parser reliably extracted structured data such as candidate details and skills (Fig. 7). The recommender ranked jobs effectively based on cosine similarity, outperforming baseline keyword searches in preliminary user studies.

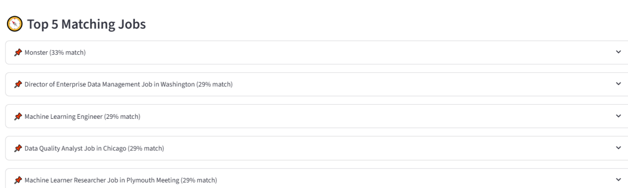

Fig. 5. Recommendation made by the system.

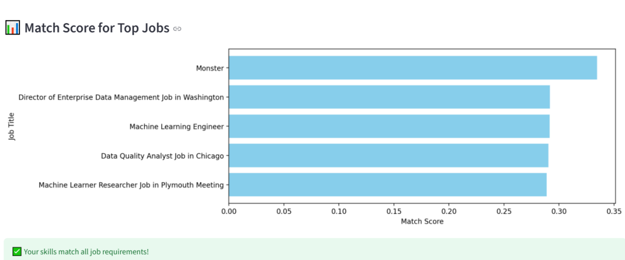

Fig. 6. Job matching visualization.

Fig. 7. Parsing Data from the Resume.

Visualization modules (Fig. 8) clearly displayed skill gaps, providing actionable feedback. User feedback indicated improved understanding of job fit and required skills.

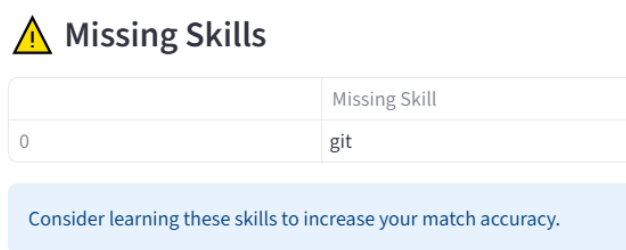

Fig. 8. Missing Skills.

Limitations included occasional noise in resume texts due to formatting inconsistencies and incomplete job descriptions. Future iterations will focus on handling such noise robustly.

Quantitative evaluation (precision@5 and recall) is planned as dataset size increases.

V. Conclusions and Future Work

We designed and implemented a modular, scalable job recommendation system integrating secure authentication, automated resume parsing, ML-driven semantic job matching, and skill gap visualization. The system improves relevance and transparency over traditional keyword-based approaches and provides users with actionable insights for skill development.

Future directions include:

- Incorporating deep contextual embeddings such as BERT to capture richer semantic relations.
- Adding collaborative filtering and hybrid models to leverage user interaction data.
- Deploying the system on cloud and big data platforms for scalability.
- Enhancing visualization with interactive, real-time dashboards.
- Expanding evaluation with larger datasets and comprehensive metrics.

Acknowledgment

The authors express their sincere gratitude to Md. Abu Rumman Refat, Faculty of Computer Science and Engineering, for his valuable guidance, constructive feedback, and encouragement.

References

[1] B. D. Puspasari, L. L. Damayanti, A. Pramono and A. K. Darmawan, "Implementation K-Means Clustering Method in Job Recommendation System," 2019 5th Int. Conf. on Big Data and Information Analytics (BigDIA).

[2] Z. Chen, W. Liang, X. Gao, Z. Zhou and M. Wu, "Research on the Accurate Recommendation Management System for Employment of College Graduates on Hadoop," 2016 Eighth Int. Conf. Adv. Comput.

[3] K. Appadoo, M. Soonnoo, and Z. Mungloo-Dilmohamud, "JobFit: Job Recommendation using Machine Learning and Recommendation Engine," 2020 IEEE CSDE.

[4] A. Nigam, A. Roy, H. Singh and H. Waila, "Job Recommendation through Progression of Job Selection," 2019 IEEE 6th Int. Conf. on Cloud Computing and Intelligence Systems (CCIS).

[5] N. D. Almalis, G. A. Tsihrintzis, N. Karagiannis and A. D. Strati, "FoDRA — A new content-based job recommendation algorithm for job seeking and recruiting," 2015 6th Int. Conf. on Information, Intelligence, Systems and Applications (IISA).

[6] P. Yi, C. Yang, C. Li and Y. Zhang, "A job recommendation method optimized by position descriptions and resume information," 2016 IEEE IMCEC.

[7] M. Jianjun, "Research on collaborative filtering recommendation algorithm based on user behavior characteristics," 2020 ICBASE.

[8] Y. Zhang, C. Yang and Z. Niu, "A Research of Job Recommendation System Based on Collaborative Filtering," 2014 ISCID.

[9] S. Li, Y. Chuancheng, W. Hongguo and D. Yanhui, "An Employment Recommendation Algorithm Based on Historical Information of College Graduates," 2018 ITME.

[10] B. Patel, V. Kakuste and M. Eirinaki, "CaPaR: A Career Path Recommendation Framework," 2017 IEEE BigDataService.

[11] V. Yadav, U. Gewali, S. Joshi, A. Joshi, and D. Wu, "SmartJob: A Smart Job Recommender System using Deep Learning," 2019 IEEE.

[12] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," 2018.