



International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 3, March 2025

Enhancing Job Search: An NLP-Based Recommender System for Job Seekers and Recruiters

Rajan Patel, Prof. Rajesh Ishwar

Department of Computer Science & Engineering, Parul University, Vadodara, Gujarat, India

ABSTRACT: For over a decade, online job boards have helped job seekers and employers connect. Traditional search methods often overwhelm job seekers with too many results, requiring time to filter them. Recruitment is crucial for organizations, requiring skills to match candidates with jobs. This article proposes a two-way recommendation system to assist both recruiters and job seekers. Using machine learning, it improves job matching by analyzing text content and similarity scores, focusing on job listings from major Saudi Arabian cities. The system integrates explicit and implicit job data, evaluated through decision support metrics, showing improved prediction accuracy.

KEYWORDS: Information System Management, Bi-Directional Recommender System, Natural Language Processing, Content-Based Filtering, Django-based application.

I. INTRODUCTION

Frictional unemployment arises from informational imperfections, including the costs of gathering and processing data, asymmetry between job providers and seekers, and cognitive limitations. Despite high unemployment, some positions remain unfilled due to these gaps.

Job seekers traditionally submit resumes to multiple companies, with success depending on resume quality. Online job platforms like LinkedIn and Indeed have transformed recruitment, yet challenges persist due to differences in language between job postings and resumes. Standardized skills can help but don't fully resolve mismatches, as skills evolve slower than job market demands. Additionally, resumes and job postings include factors like location, salary, and career growth, which vary by industry.

Personalized, reciprocal job recommendation systems can address these issues by improving job matching and reducing frictional unemployment. This research aims to:

- Help job seekers align skills with market demand.
- Assist recruiters in streamlining hiring.
- Improve job-candidate matching accuracy.
- Optimize digital recruitment.

Figure 1: Illustrates the workflow of the bi-directional recommender system. The process begins with job listings and user profile data as inputs. Pre-processing and training steps generate candidate profiles and job representations. Finally, a matching algorithm produces tailored job recommendations for job seekers and suitable candidate recommendations for employers.

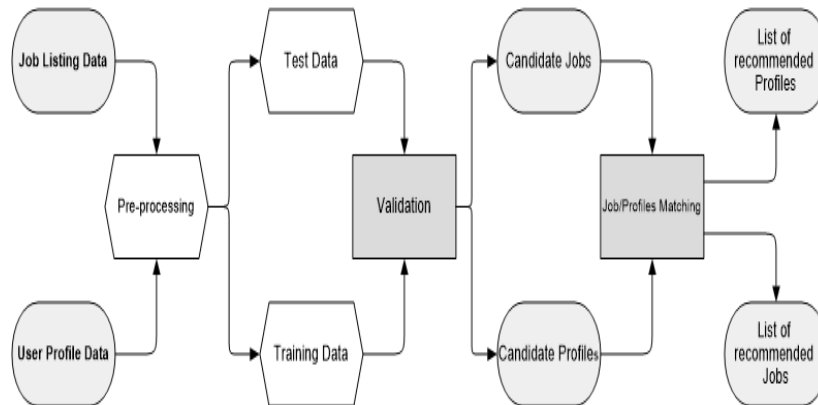


Figure 1. Bi-directional recommender system flowchart.

II. RELATED WORK

Recommender systems help users filter information efficiently. Collaborative filtering (CF) suggests items by identifying users with similar preferences, commonly used in online bookstores and recruitment to expand candidate pools. The expectation-maximization (EM) algorithm improves bi-directional CF by integrating user and item ratings. Content-based filtering (CBF) relies on user-provided data to build models based on past preferences, which we applied in our study.

While CF and CBF have static interactions, sequential recommendation systems (SRS) track dynamic user behavior using deep bidirectional self-attention models to predict preferences. Conversational recommender systems (CRS) allow real-time, natural language interactions for personalized recommendations. Each approach—CF, CBF, SRS, or CRS—has unique structures and behaviors, requiring careful selection based on the use case and data characteristics.

III. PROPOSED SYSTEM

A. System Overview

The proposed system facilitates a convenient and automated job matching process through a web-based application. The primary objectives of this system are as follows:

1. **Reciprocal Job Matching:** Develop an intelligent platform to connect job seekers with most suitable opportunities based on their skills, interests, and qualifications.
2. **Higher Recommendation Accuracy:** Employ advanced machine learning and natural language processing (NLP) techniques to compare job descriptions and resumes for higher context-based relevance.
3. **Minimizing Frictional Unemployment:** Streamline the recruitment process by reducing the time and energy that job applicants spend finding right opportunities.
4. **User Interaction and Feedback Analysis:** feedback mechanisms add to optimize the algorithm efficiency and increase user satisfaction.

B. Target User

1. **Job Seekers:** Job seekers look for opportunities that match their skills, experience, and career goals.
2. **Recruiters:** Hiring managers or companies searching for the right candidates to fill open positions.

3. Human Resource Managers: HR professionals who want to make hiring easier by cutting down the time spent reviewing resumes.
4. Career Advisors: Experts who assist job seekers in finding suitable opportunities and enhancing their chances of employment.
 - a. System Architecture

System architecture refers to the internal structure of a system, its components, their interrelations, and the rules that govern their functioning. It is a high-level plan that describes how hardware, software, networks, and data interact to provide desired functionalities. A good system architecture promotes scalability, efficiency, security, and maintainability. It is usually depicted in the form of diagrams and models that show component interactions, data flow, and integration points. In areas like software development and IoT, system architecture is an essential aspect in crafting resilient and efficient solutions.

Component	Interaction with Server	Data Exchanged
Sign-Up Page	Sends registration data	Name, email, password, skills, resume
Log-In Page	Sends credentials	Email, password
View Profile (Job Seeker)	Displays and manages profile	Name, email, skills, experience, resume
Recommended Jobs (Job Seeker)	Fetches personalized job listings	User skills, preferences, job data
Current Job Posting (Recruiter)	Manages and updates job listings	Job title, description, requirements, location
Eligible Candidates Page (Recruiter)	Fetches list of candidates	Candidate details, skills, experience

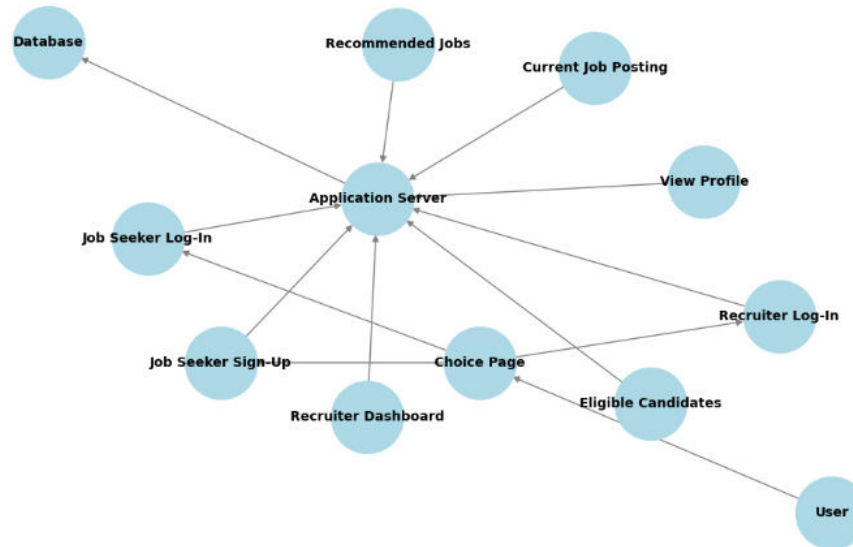


Figure.2 System Architecture

C. Functional Description

1) User Interface (Job Seeker Side)

Job Seeker Registration & Authentication: Job seekers register with an email and password, providing essential details such as name, contact number, and experience. Secure login includes email verification and hashed passwords. Forgotten passwords can be recovered securely, and Multi-Factor Authentication (MFA) enhances security. Session management ensures safe access and protects accounts.

Profile Management: Users can update their profiles with work skills, experience, education, and interests. They can upload CVs and include portfolio or social media links. A completeness counter increases visibility. Real-time data storage prevents data loss.

Job Search & Recommendations: The platform offers personalized job recommendations based on user experience, qualifications, and job application history. Advanced filters help search by location, sector, salary, and employment type. Machine Learning improves job matching accuracy. Users can save job postings and receive notifications for relevant openings. Direct recruiter interaction is available via email.

Interview & Career Counsel: Features include interview scheduling, AI-driven resume feedback, and mock interview practice. Career guides, articles, and video tutorials support career development. Personalized reminders notify users about skill-building workshops and certification courses.

Application Tracking & Management: Applicants can track their applications categorized as rejected, viewed, shortlisted, or pending. Automated reminders highlight missing information and deadlines. Recruiters provide real-time feedback on applications.

2) User Interface (Recruiter Side)

Recruiter Registration & Authentication: Recruiters register using company details and authentication documents. The system verifies company authenticity before granting access. Secure login includes two-factor authentication. A dashboard summarizes job postings and candidate interactions.

Job Posting & Management: Recruiters create job postings with details such as title, description, qualifications, salary, and location. Expired listings are removed automatically. Analytics track job post performance based on engagement and application rates.

Candidate Search & Shortlisting: ML-powered recommendations identify top candidates based on skills, experience, location, and education. Shortlisting software segregates candidates, and automated ranking identifies the best fits. Recruiters can communicate directly with shortlisted candidates via email.

Application Review & Interview Scheduling: Recruiters can track applications categorized as new, under review, shortlisted, or rejected. Built-in scheduling simplifies interview coordination, sending automated reminders to both recruiters and candidates.

Recruiter Analytics & Reporting: Real-time insights provide job performance data, candidate application trends, and hiring success rates. Detailed reports help assess recruitment efficiency. Engagement tracking tools analyze job post effectiveness. ML-driven recommendations optimize job descriptions.

3) User Interface (Admin Side)

Admin Section & Security Management: Admins log in securely with MFA. They monitor user activity, login attempts, and suspicious behavior. Regular security audits ensure compliance with data protection policies.

User Management: Admins oversee job seeker and recruiter accounts, modifying or deactivating users as needed. Role-based access control prevents unauthorized changes. Users violating policies can be banned or suspended, receiving automatic notifications.

Job Posting & Content Moderation: Admins review job listings to ensure policy compliance. Suspicious posts are flagged for human review. ML moderation tools detect misleading job descriptions and spam.

Platform Performance & Data Analysis: The admin panel provides insights into user interactions, job postings, and recruiter activities. Reports can be exported in various formats (CSV, PDF). Visual tools such as charts and heatmaps facilitate data interpretation. Predictive analytics help forecast hiring trends.

Customer Support & Issue Resolution: Admins manage support tickets via an integrated help desk. AI-powered chatbots handle common inquiries. Escalation procedures address complex issues. User feedback improves platform usability.

D. Data Management

User Preferences & Accounts: Stores registration data, login details, and job search preferences. Tracked behavior improves recommendation accuracy and search relevance.

Job Posting Information: Maintains records of job listings, including title, company, location, salary, and required skills. The system updates listings based on employer preferences and market demand.

Resume Management: Uses NLP to extract key details from resumes, categorizing candidates based on skills, experience, and education.

Application Tracking: Logs job application statuses (submitted, in review, shortlisted, rejected). Applicants can track progress, receive updates, and schedule interviews.

Recommendation Engine: Uses collaborative and content-based filtering to generate targeted job recommendations. Performance is optimized based on accuracy, precision, and recall metrics.

User Feedback & Ratings: Collects feedback and ratings from job seekers regarding employers and job listings. Sentiment analysis enhances job listing quality and employer credibility.

F. Implementation Details

1) Technology Used:

- Frontend:
 - **Languages:** HTML, CSS, JavaScript, Django Templates
- Backend:
 - **Languages:** Python
 - **Framework:** Django

- Database:
- PostgreSQL
- Machine Learning & Data Processing:
- Scikit-Learn
- Natural Language Processing (NLP)
- Pandas & NumPy
- Cloud Platform/Hosting:
- Docker
- Other Tools:
- **IDE/Code Editor:** Visual Studio Code, Jupyter Notebook

Development Process: The Job Recommendation System followed an Agile-inspired development process, emphasizing flexibility, iterative improvements, and stakeholder collaboration. While not strictly adhering to a specific Agile framework, the project incorporated key Agile principles:

- **Iterative Development:** The system was built in rapid iterations, each focusing on distinct features like user authentication, job posting, recommendations, and recruiter functionalities. This incremental approach enabled early testing and smooth development.
- **Evolving Requirements:** Features such as resume parsing, candidate-job matching, and recruiter dashboards evolved based on user feedback, enhancing system usability and precision.
- **Continuous Integration:** Frequent commits to GitHub ensured smooth code integration and early identification of issues. The development team leveraged Django ORM for efficient database management and backend stability.

This Agile-driven approach allowed for a dynamic and adaptable development process, improving system efficiency over time.

1) **Challenges we faced:** During the development of the travel package booking website, several technical challenges were encountered. Each challenge was systematically addressed to ensure system reliability, data integrity, and user experience enhancement. The key challenges and their corresponding solutions are outlined below:

1. Natural Language Processing (NLP) Complexity:

Extracting meaningful insights from job descriptions and resumes was a complex task due to the unstructured nature of textual data. Different formatting styles, synonym variations, and context-specific language in resumes and job postings made information extraction challenging. To overcome this, we employed advanced NLP techniques such as Named Entity Recognition (NER) and word embeddings using models like Word2Vec and BERT.

2. Matching Algorithm Accuracy:

Developing a precise job recommendation system required balancing multiple factors, including user preferences, job relevance, and skill matching. Initially, the system provided suboptimal matches due to oversimplified filtering mechanisms. To enhance accuracy, we experimented with different machine learning models, including content-based filtering, collaborative filtering, and hybrid approaches. We calculated similarity scores between job descriptions and resumes using TF-IDF vectorization and cosine similarity, ensuring better alignment between candidates and job postings. Additionally, we incorporated user feedback loops where candidates could rate recommendations, allowing the system to learn and refine its suggestions over time.

3. Model Creation:

Building an effective machine learning model for job recommendations involved multiple iterations of training and testing. Choosing the right model architecture and optimizing hyperparameters took considerable effort. We explored deep learning techniques such as neural networks for feature extraction and logistic regression for classification. To improve the model's performance, we utilized large-scale job datasets and implemented techniques like feature selection, hyperparameter tuning, and cross-validation. Regular evaluation using metrics like precision, recall, and F1-score helped us identify weaknesses in the model and improve its performance through iterative refinements.

G. System Diagram

1) Use Case Diagram: A Use Case Diagram graphically represents user interactions with the Job Recommendation System, defining key functionalities. In this diagram:

- Job Seekers can register, log in, update their profiles, search for jobs, apply for jobs, and receive job recommendations
- Recruiters can register, log in, post job listings, search for candidates, shortlist applicants, and manage job postings.
- Admins manage users, monitor system performance, and handle security.

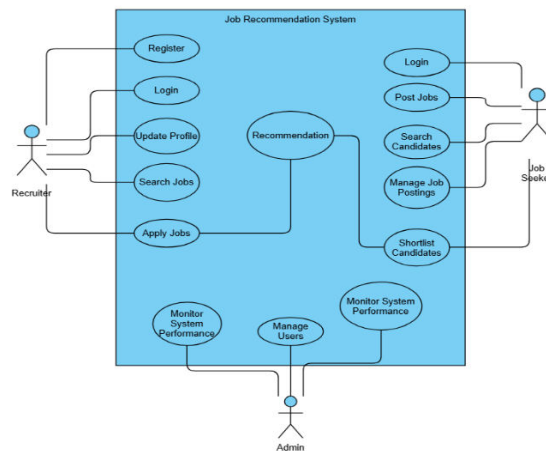


Figure.3 Use Case Diagram

1. DFD Diagram:

a) Context Level (Level 0): Level 0 DFD provides an overall view of the Job Recommendation System, illustrating interactions with Job Seekers, Recruiters, Admins, and the Database for job searching, job posting, user management, and data storage.

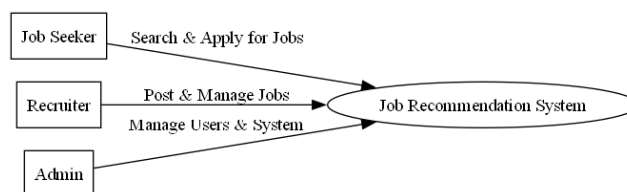


Figure.4 DFD Level 0 Diagram

b) Level 1 DFD Diagram: The Level 1 DFD expands on the Job Recommendation System, detailing its internal processes. It includes:

- Job Seeker Management for registration, login, profile updates, and job applications.
- Recruiter Management for job postings, candidate searches, and shortlisting applicants.
- Job Recommendation Engine for analyzing job seeker profiles and providing personalized job recommendations.
- User Authentication & Security for managing secure login, access control, and data protection.

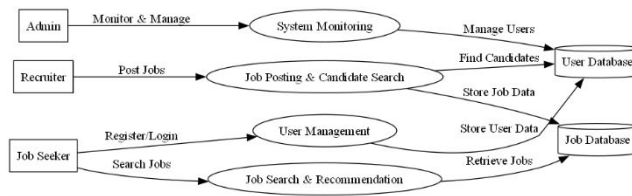


Figure.5 DFD Level 1 Diagram

IV. RESULTS

A. Recommendation System Evaluation

To validate our predictions, we utilized a confusion matrix to assess both correct and incorrect classifications while also gaining insights into the types of errors made.

The most accurate recommendations appear along the diagonal of the confusion matrix (see Figure 6).



Figure 6. Confusion matrix

The results indicate:

- All five resumes for the PHP Developer job profile were correctly classified, demonstrating excellent accuracy.
- Similarly, all five resumes for the Data Scientist job profile were accurately identified, which is highly satisfactory.
- For the Data Engineer job profile, only two out of five resumes were correctly classified, likely due to the overlap in skills between related roles.
- Four out of five resumes were correctly classified under the Java Developer profile, which is a strong result.
- Likewise, four out of five resumes were correctly assigned to the Back-end Developer job profile, showing a high level of accuracy.

Overall, the model correctly classified 20 out of 25 resumes, indicating strong performance. Additionally, we computed key evaluation metrics such as precision, recall, F1-score, and accuracy to further analyze the classification results.

B. Testing and Validating the Model

A predictive model's accuracy is measured by the percentage of correct predictions, calculated as:

$$\text{Accuracy} = \frac{\text{Number of Correct Prediction}}{\text{Total prediction}}$$

Since this model handles multi-class classification, the evaluation steps include:

- Obtaining predictions.

- Counting correct predictions.
- Computing accuracy.
- Analyzing results.
- Higher accuracy indicates better performance, with 1 being perfect and 0 being entirely incorrect.

Model Performance

Entity	Accuracy	Precision	Recall	F1-Score
Name	99.88%	0.9989	0.9989	0.9987
Location	99.77%	0.9977	0.9977	0.9977
Skills	99.08%	0.9895	0.9908	0.9898
Education	100%	1.0000	1.0000	1.0000

These high scores indicate the model's effectiveness in identifying Name, Location, Skills, and Education entities.

C. Sign-Up Page

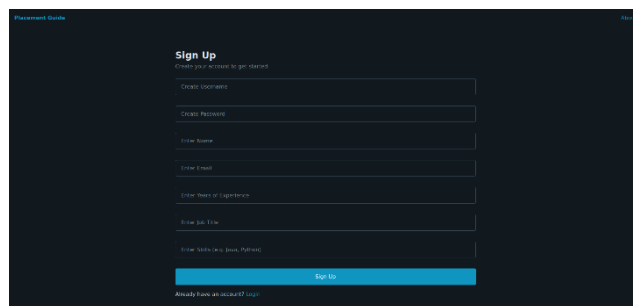


Figure 7. Sign-up Page

Figure 7 presents the Sign-Up Page, where new users register by providing their credentials such as name, email, mobile number, and password. This step ensures secure access and personalized recommendations based on their profile information.

D. Log-In Page

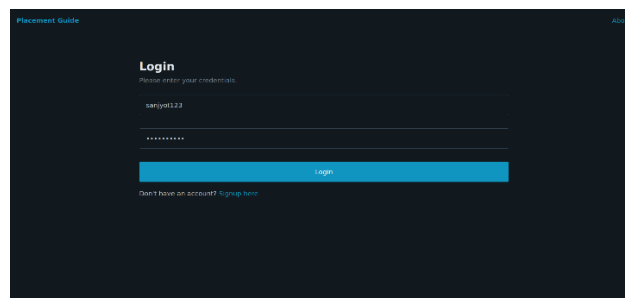


Figure 8. Login Page

Figure 8 illustrates the Job Seeker Log-In Page, allowing registered users to access their accounts. By entering their email and password, job seekers can view job recommendations, update their profiles, and apply for relevant positions.

E. Profile Page

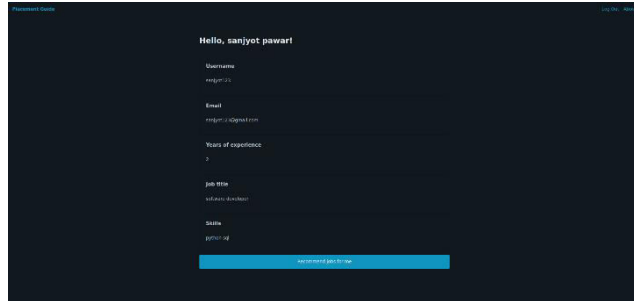


Figure 9. Profile Page

Figure 9 depicts the Profile Page, where job seekers can enter and manage their personal and professional details, including skills, experience, education, and resume uploads. This data plays a crucial role in improving job recommendations and matching them with suitable opportunities.

F. Recommendation Page

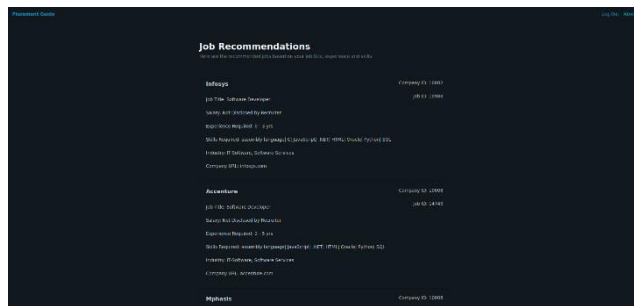


Figure 10. Recommendation Page

Figure 10 showcases the Recommendation Page, which provides personalized job suggestions based on the job seeker's profile, skills, and preferences. The system utilizes machine learning algorithms to analyze job postings and recommend the most relevant positions.

G. Job Posting Page

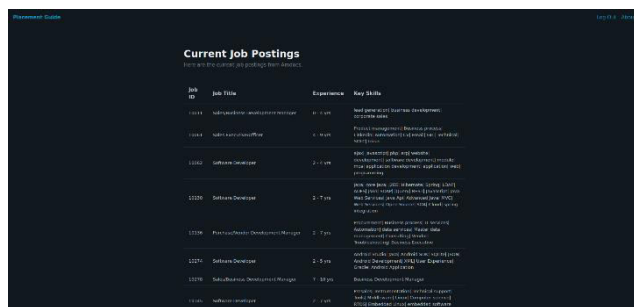


Figure 11. Job Posting

Figure 11 illustrates the Job Posting Page, where recruiters can create and manage job listings by specifying details such as job title, description, required skills, and application deadlines. This page ensures structured job postings to attract the most relevant candidates.

H. Eligible Candidates Page

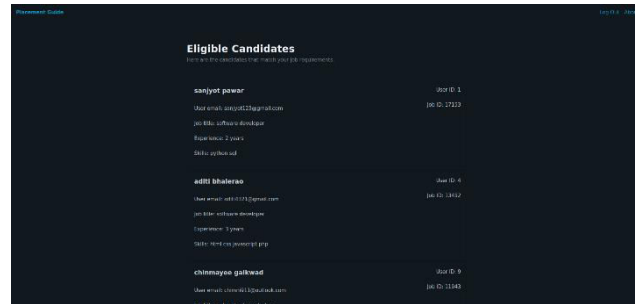


Figure 12. Eligible Candidates

Figure 12 presents the Eligible Candidates Page, which helps recruiters filter and view job seekers who meet the specified job requirements. The system uses machine learning techniques to rank candidates based on their profile match, experience, and skills, aiding in efficient hiring decisions.

V. CONCLUSION

The proposed NLP-based recommender system enhances job matching using machine learning and natural language processing. It streamlines job searches for seekers and helps recruiters identify suitable candidates, reducing frictional unemployment.

Machine learning ensures accurate recommendations by analyzing resumes and job postings, evaluated through metrics like precision, recall, and F1-score. Using NLP techniques such as named entity recognition and semantic similarity, the system extracts key information to provide tailored job suggestions.

Tested on Saudi Arabian job postings, the system effectively identifies in-demand skills and recommends relevant roles. Recruiters benefit from automated filtering, reducing manual errors and improving hiring efficiency.

Addressing skill mismatches, the recommendation engine aligns candidates with job listings based on explicit and implicit details. It continuously refines suggestions using job market trends and user preferences.

Feedback mechanisms enhance accuracy, allowing users to rate recommendations, while sentiment analysis helps refine the process. These improvements ensure the platform adapts to evolving job markets and remains effective.

VI. FUTURE WORK

While the NLP-based job recommender system shows promising results, several improvements can enhance its effectiveness. Refining experience timeline analysis will ensure outdated work histories don't affect job matches, while leveraging industry-specific ontologies will improve automated skills detection, bridging gaps between how skills are listed by job seekers and employers.

Geo-location integration will allow users to receive job recommendations based on preferred locations, benefiting both job seekers and recruiters. Advanced deep learning models like BERT or GPT can further enhance job description and resume analysis, capturing nuances in job roles and qualifications.

Real-time feedback mechanisms will refine recommendation accuracy by incorporating user ratings, enabling continuous system improvement. Expanding the dataset beyond Saudi Arabia to include diverse industries and global job markets will increase robustness. These enhancements will make the system more efficient and intelligent for job seekers and recruiters.

REFERENCES

- [1] Catherine, R.; Visweswariah, K.; Chenthamarakshan, V.; Kambhatla, N. PROSPECT: A system for screening candidates for recruitment. [CrossRef]
- [2] Parida, B.; KumarPatra, P.; Mohanty, S. Prediction of recommendations for employment utilizing machine learning procedures and geo-area based recommender framework. *Sustain. Oper. Comput.* 2022, 3, 83–92. [CrossRef]
- [3] Kokkodis, M.; Ipeiritos, P.G. Demand-Aware Career Path Recommendations: A Reinforcement Learning Approach. *Manag. Sci.* 2021, 67, 4362–4383. [CrossRef]
- [4] Lacic, E.; Reiter-Haas, M.; Kowald, D.; Dareddy, M.R.; Cho, J.; Lex, E. Using autoencoders for session-based job recommendations. *User Model. User Adapt. Interact.* 2020, 30, 617–658. [CrossRef]
- [5] Saeed, T.; Sufian, M.; Ali, M.; Rehman, A.U. Convolutional Neural Network Based Career Recommender System for Pakistani Engineering Students. In *Proceedings of the 2021 International Conference on Innovative Computing (ICIC), Lahore, Pakistan, 9–10 November 2021*; pp. 1–10. [CrossRef]
- [6] Zhu, G.; Chen, Y.; Wang, S. Graph-Community-Enabled Personalized Course-Job Recommendations with Cross-Domain Data Integration. *Sustainability* 2022, 14, 7439. [CrossRef]
- [7] Jannach, D.; Manzoor, A.; Cai, W.; Chen, L. A Survey on Conversational Recommender Systems. *ACM Comput. Surv.* 2021, 54, 1–36. [CrossRef]
- [8] Syed, M.H.; Huy, T.Q.B.; Chung, S. Context-Aware Explainable Recommendation Based on Domain Knowledge Graph. *Big Data Cogn. Comput.* 2022, 6, 11. [CrossRef]
- [9] Wu, L. Collaborative Filtering Recommendation Algorithm for MOOC Resources Based on Deep Learning. *Complexity* 2021, 2021, 5555226. [CrossRef]
- [10] Sun, N.; Chen, T.; Guo, W.; Ran, L. Enhanced Collaborative Filtering for Personalized E-Government Recommendation. *Appl. Sci.* 2021, 11, 12119. [CrossRef]



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details