

## Skill Career Explorer

Dr M Vinod<sup>1</sup>, Mariyala Gayathri<sup>2</sup>, Bommana Greeshma Reddy<sup>3</sup>, Benolla Lahari<sup>4</sup>, Puli Madhuri<sup>5</sup>

<sup>1</sup>Associate Professor; Department Of Computer Science And Engineering , Bhoj Reddy Engineering College For Women, Hyderabad, India.

<sup>2,3,4,5</sup>B.Tech Students; Department Of Computer Science And Engineering , Bhoj Reddy Engineering College For Women, Hyderabad, India.

Mail Id; [mariyalagayathri2@gmail.com](mailto:mariyalagayathri2@gmail.com)<sup>2</sup>, [bommanagreeshma@gmail.com](mailto:bommanagreeshma@gmail.com)<sup>3</sup>, [benollalahari1623@gmail.com](mailto:benollalahari1623@gmail.com)<sup>4</sup>, [pullimadhuri23@gmail.com](mailto:pullimadhuri23@gmail.com)<sup>5</sup>

### Abstract

*Skill Career Explorer is an AI-driven web application designed to deliver intelligent and personalized career guidance for students and job seekers. In an increasingly competitive employment landscape, individuals often face challenges in selecting suitable career paths, understanding industry expectations, and acquiring relevant skills. The proposed system addresses these challenges by bridging the gap between academic knowledge and professional requirements through a comprehensive, data-centric career planning platform.*

*The application integrates multiple functional modules, including Resume Analysis, Skill Gap Identification, Career Recommendation, Roadmap Generation, and Career Comparison. The Resume Analysis module employs Natural Language Processing techniques to extract relevant skills and competencies from user resumes. These extracted skills are then evaluated against industry benchmarks in the Skill Gap Identification module to determine missing or underdeveloped competencies. Based on this analysis, the system generates personalized career recommendations along with guidance on certifications, internships, and skill development opportunities.*

*Furthermore, the platform creates structured learning roadmaps that provide step-by-step guidance for users to enhance their abilities and achieve targeted career objectives. Progress tracking and visualization features are incorporated to help users monitor their development over time. By combining artificial intelligence with a user-centric interface, Skill Career Explorer supports informed decision-making, improves employability, and enhances overall career readiness.*

**Keywords**— Skill Career Explorer, Artificial Intelligence, Career Guidance System, Natural Language Processing, Resume Analysis, Skill Gap Identification, Career Recommendation, Learning Roadmap Generation, Employability Enhancement, Data-Driven Decision Making.

### Introduction

Skill Career Explorer is an AI-driven career guidance application designed to assist students and

job seekers in identifying suitable career paths and improving employability. The system analyzes user skills, evaluates resumes, detects missing competencies, and recommends appropriate career options. In addition, it generates structured learning roadmaps and provides comparison tools that support effective career decision-making. The application consists of multiple integrated modules including Resume Analysis, Skill Gap Analysis, Career Recommendation, Roadmap Generation, Career Comparison, and Progress Tracking. These modules collectively help users understand their current skill level and guide them toward achieving their professional goals.

Traditional career guidance methods primarily rely on generic job portals, manual resume preparation, and static advisory platforms. These approaches mainly focus on job listings rather than skill development and provide limited personalization. Most existing systems do not analyze user profiles comprehensively, resulting in ineffective career guidance. They lack automated resume analysis, structured learning roadmaps, and meaningful career comparison features. Students and job seekers therefore struggle to identify suitable career paths, understand industry requirements, recognize missing competencies, and plan structured learning strategies. These challenges ultimately lead to inefficient career planning and reduced employability.

The proposed Skill Career Explorer system provides an integrated AI-based solution for career guidance. It utilizes Natural Language Processing techniques to analyze resumes, compares user skills with industry requirements, and generates personalized recommendations. The system includes resume parsing, skill gap detection, AI-based career recommendations, structured roadmap generation, career comparison tools, and progress tracking with visualization. This approach ensures that users receive personalized insights and continuous guidance throughout their career development process. The advantages of the proposed system include personalized career guidance, improved employability, structured learning paths, intelligent skill analysis, data-driven decision-making, user-friendly interface design, and scalable architecture.

### Requirement Analysis

The Skill Career Explorer system requires several functional and non-functional components to operate efficiently. The functional requirements include user registration and authentication, resume upload and analysis, skill extraction and evaluation, skill gap detection, career recommendation, roadmap generation, career comparison, and progress tracking. These features ensure that users can interact with the system seamlessly and receive accurate career guidance.

Non-functional requirements focus on system quality attributes such as performance, scalability, usability, maintainability, and security. The system must provide fast response times for resume analysis and recommendations while supporting multiple concurrent users. A user-friendly interface is required to ensure ease of use, and secure data handling mechanisms must protect user profiles and resumes. Additionally, the system should maintain a modular architecture to allow easy updates and enhancements.

Hardware requirements include an Intel Core i5 processor or higher, a minimum of 8 GB RAM, and at least 512 GB storage, with support for Windows or Linux operating systems. Software requirements include HTML, CSS, and JavaScript for the frontend, Node.js with Express.js for the backend, MongoDB for database management, and Python for implementing NLP modules. Libraries such as Scikit-learn and other NLP frameworks are used to enhance system intelligence and performance.

### System Design

The Skill Career Explorer system follows a three-tier architecture consisting of frontend, backend, and database layers. The frontend handles user interaction and interface components, while the backend processes business logic, resume analysis, and recommendation algorithms. The database stores user information, resumes, skills, and career data. This architecture ensures scalability, maintainability, and efficient communication between components.

Users interact with the system by registering, uploading resumes, and performing skill analysis. The system extracts skills, identifies missing competencies, and generates career recommendations. Users can then generate learning roadmaps, compare multiple career options, and track their progress. The administrator manages career data, updates skill requirements, and monitors system performance to ensure accurate recommendations.

The system design includes several classes such as User, Resume, Skill, Career, Recommendation, Roadmap, Analysis, and Admin. These classes interact with one another to provide personalized career guidance. The User class manages profile data and interactions, the Resume class handles resume processing, the Analysis class performs skill

evaluation, and the Recommendation class generates career suggestions. The Roadmap class creates structured learning plans, while the Admin class maintains system data.

### Implementation

The implementation of Skill Career Explorer utilizes multiple algorithms to perform intelligent analysis and recommendations. Resume parsing is performed using NLP techniques to extract relevant skills from user resumes. Skill gap detection is implemented using a set difference approach that compares user skills with required skills for specific career roles. Career recommendations are generated using content-based filtering, which considers user skills, interests, and academic background. Roadmap generation uses rule-based logic to assign learning paths and suggest courses, certifications, and timelines. Career comparison is performed using multi-criteria decision analysis, which evaluates career options based on salary, skills required, growth opportunities, and difficulty level.

The system is divided into modules including Resume Analysis, Skill Gap Detection, Career Recommendation, Roadmap Generation, and Career Comparison. Each module is implemented independently and integrated through APIs to ensure modularity and scalability.

### Testing

Software testing for the Skill Career Explorer project plays a crucial role in ensuring that every component of the system operates correctly, efficiently, and securely. As the platform is an AI-driven career guidance application that handles sensitive information such as resumes, skill sets, and user preferences, validating system reliability is essential. The testing process verifies that all modules deliver accurate outputs and maintain data integrity while providing a smooth user experience. Each module is evaluated against predefined functional requirements. The Resume Analysis component is tested to confirm that it accurately extracts skills and key information from uploaded resumes using natural language processing techniques. Similarly, the Skill Gap Analysis module is validated to ensure that it compares user skills with industry requirements and correctly identifies missing competencies. The Career Recommendation module is examined to confirm that it generates suitable career options based on user skills, academic background, and interests. The Roadmap Generation feature is tested to ensure that it produces structured learning paths with courses, certifications, and timelines. Additionally, the Career Comparison functionality is evaluated to verify accurate comparisons using parameters such as salary range, growth potential, required skills, and difficulty level. The Progress Tracking module is

also tested to ensure that user development is properly reflected through charts and indicators.

Security testing is conducted to protect user authentication and safeguard confidential information such as resumes and personal details. Performance testing ensures that the application can handle concurrent users without noticeable delays. Overall, the testing phase helps identify defects, enhance system performance, and guarantee that the Skill Career Explorer platform provides reliable and user-friendly career guidance.

#### **Dimensions of Testing**

Testing was performed across multiple dimensions to ensure comprehensive quality assurance. These dimensions include application layers, scale of testing, testing types, and testing methodology. Testing across application layers covered database operations, API functionality, and user interface behavior. The scale of testing involved unit testing, module testing, integration testing, and scenario-based testing. Various testing types such as functional, performance, and security testing were implemented to validate different aspects of the system. In terms of methodology, exploratory testing, scripted manual testing, and automated testing approaches were adopted.

The Software Testing Life Cycle (STLC) forms an essential part of the Software Development Life Cycle but focuses exclusively on testing activities. STLC begins once requirements are finalized and continues through structured phases to ensure software quality. During the early stages, the testing team defines the scope, entry and exit criteria, and prepares test cases. This proactive approach reduces testing time and improves product quality. After development, test execution begins, enabling early detection and correction of defects.

#### **5 Stages of Testing**

Software testing for the system followed six major phases: requirement analysis, test planning, test case development, test environment setup, test execution, and test closure. During requirement analysis, the testing team reviewed project features such as resume upload, skill extraction, skill gap analysis, career recommendation, and roadmap generation. This helped identify testable requirements and determine expected module behavior.

In the test planning phase, a comprehensive strategy was prepared. This included selecting testing techniques, defining scope, identifying tools, allocating resources, and scheduling timelines. Responsibilities were assigned to ensure efficient execution of testing tasks.

The test case development phase involved designing detailed test scenarios with inputs, expected outputs, and execution steps. These covered resume uploads in different formats, skill gap analysis for various roles, career recommendation validation, roadmap generation, and career comparison functionalities.

During test environment setup, a realistic testing environment was configured. The frontend technologies included HTML, CSS, and JavaScript, while the backend used Node.js with Express.js and MongoDB as the database. Required NLP libraries and API testing tools were installed to simulate real-world usage conditions.

Test execution involved running all prepared test cases. Individual modules were tested first, followed by integration testing to validate smooth interaction among components. Outputs were compared with expected results, and defects were logged and fixed. The system was retested after corrections to ensure reliability.

Finally, in the test closure phase, all testing activities were completed, and results were analyzed. Reports summarizing test outcomes, detected defects, and implemented fixes were prepared. After confirming that all modules met requirements and performed efficiently, the system was approved for deployment.

#### **Types of Testing**

Black box testing was performed to evaluate system functionality without examining internal code structure. This method focused on validating inputs and outputs across various testing levels, including unit, integration, and system testing. White box testing was also conducted to analyze internal logic and code structure. This approach included techniques such as statement coverage, branch coverage, and path coverage to ensure comprehensive testing of program flow and logic.

#### **Test Cases**

Several test scenarios were executed to validate system functionality. User registration was tested using valid credentials, resulting in successful account creation and dashboard access. Duplicate registration attempts were also evaluated, ensuring that appropriate error messages were displayed. Login functionality was tested with both correct and incorrect passwords to confirm authentication reliability. Resume upload and analysis were tested using PDF and DOCX formats, verifying accurate skill extraction and summary generation. Skill gap analysis was validated by comparing user skills with selected career roles, producing correct match scores and missing skill lists. Career recommendation functionality generated appropriate career suggestions along with growth insights. Roadmap generation produced step-by-step learning plans for missing skills. Finally, career comparison testing confirmed accurate comparisons of salary, required skills, and growth opportunities between selected roles. All executed test cases produced expected outputs, confirming proper system functionality.

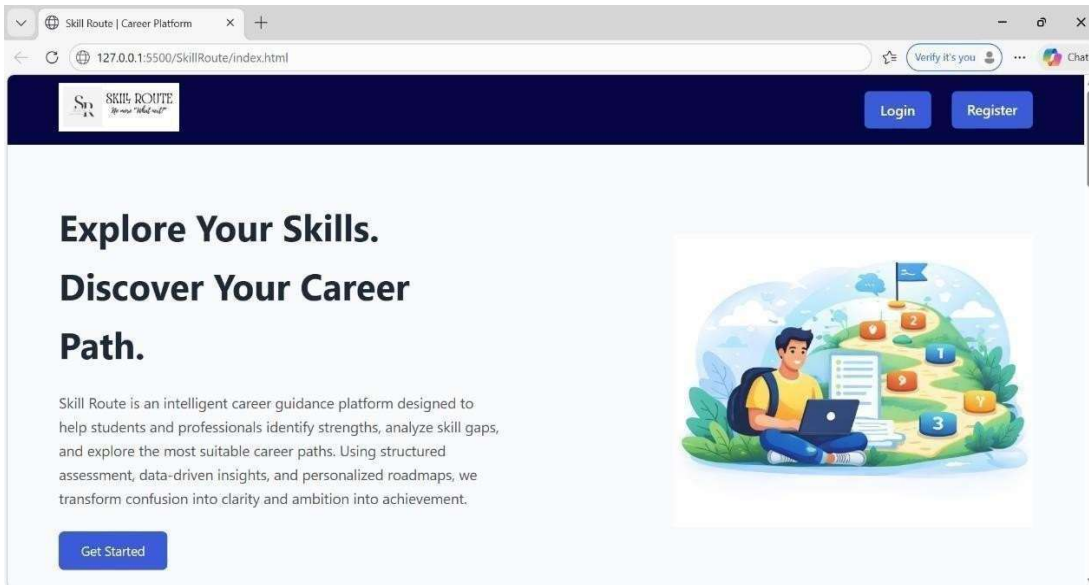
#### **Results**

The implementation results demonstrate that the system successfully extracts skills from resumes, identifies skill gaps, recommends relevant career

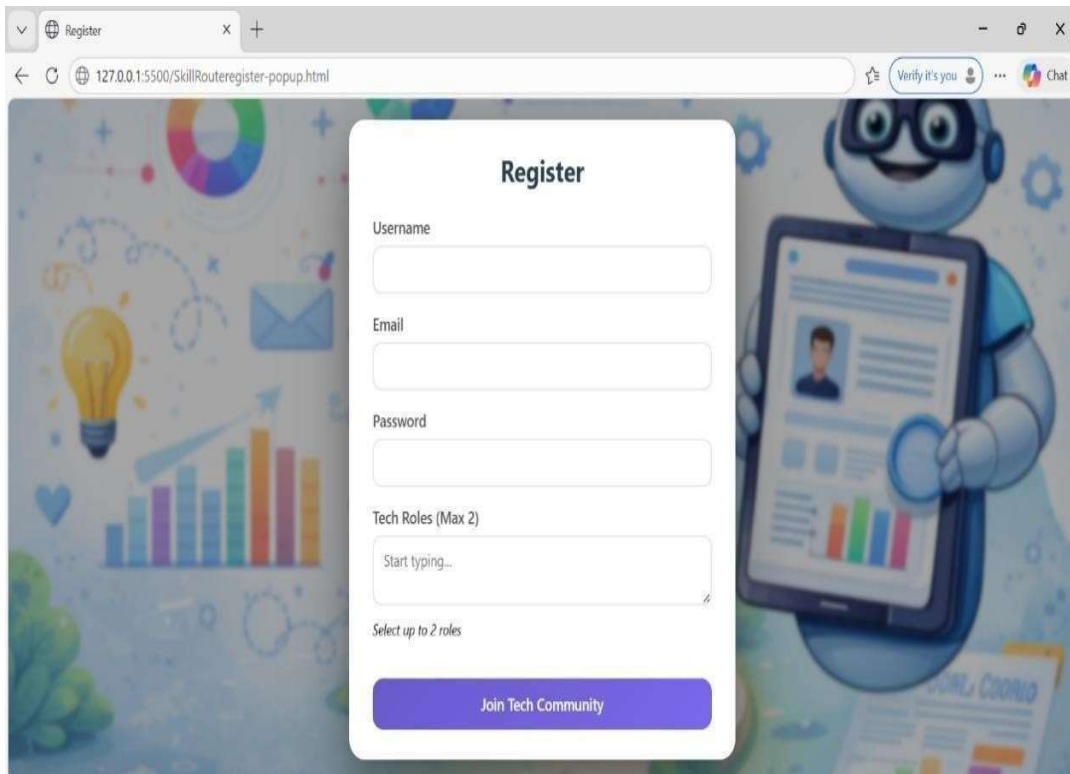
options, generates structured learning paths, and compares multiple career choices effectively. The system provides meaningful insights that help users

improve their skills and make informed career decisions.

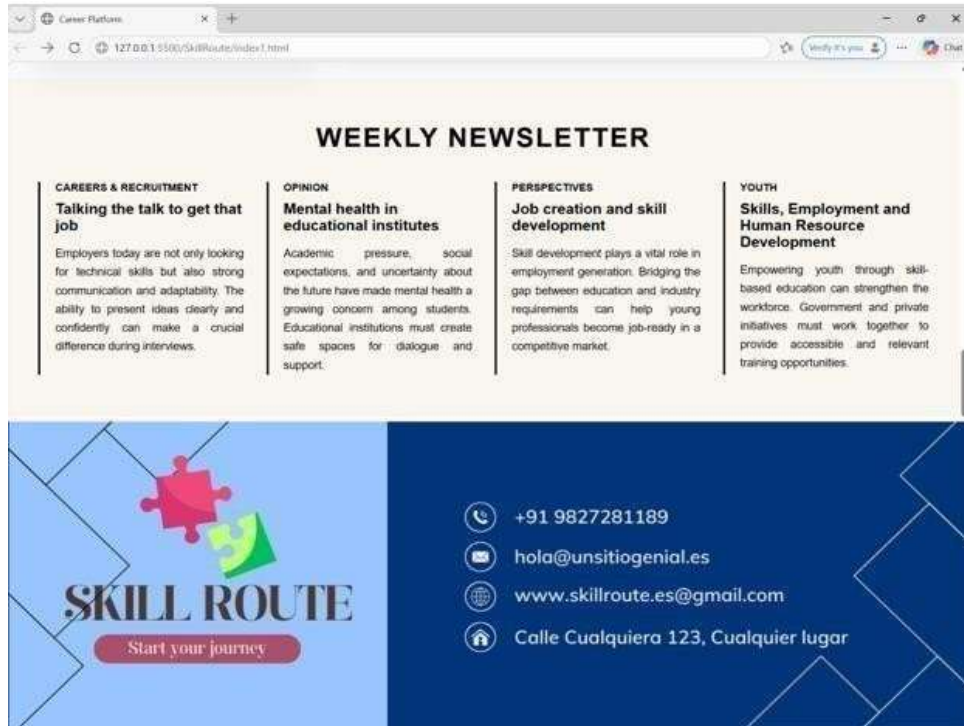
### Screenshots



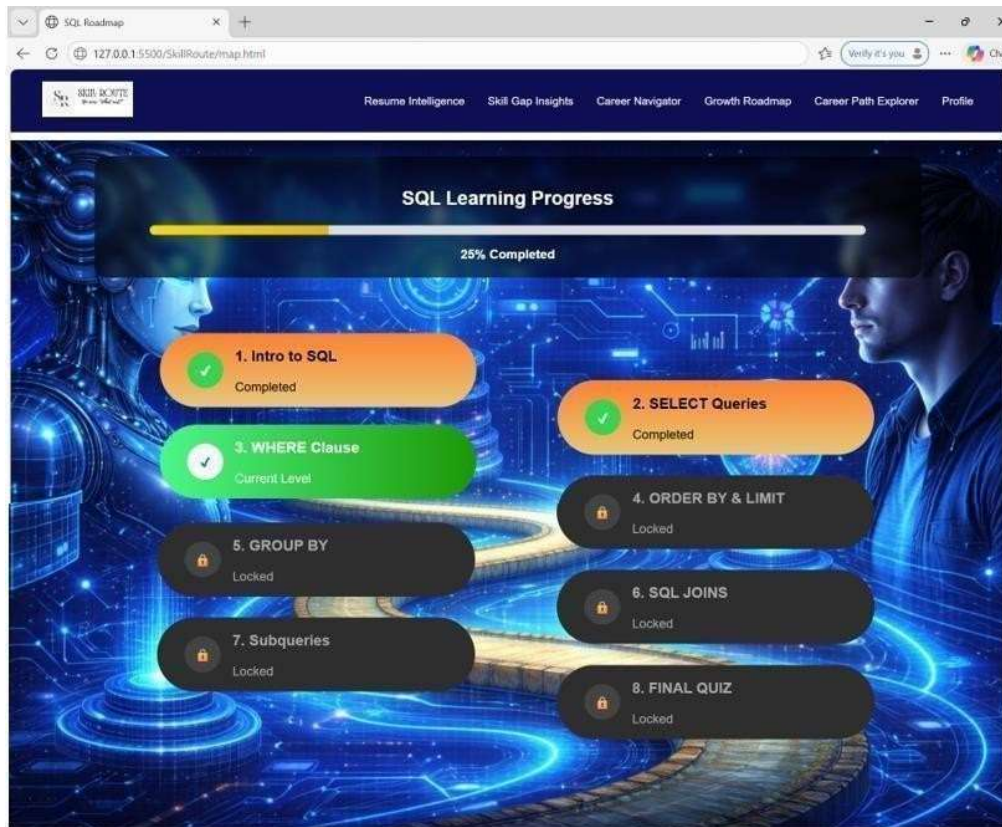
Screenshot 1 Welcome page



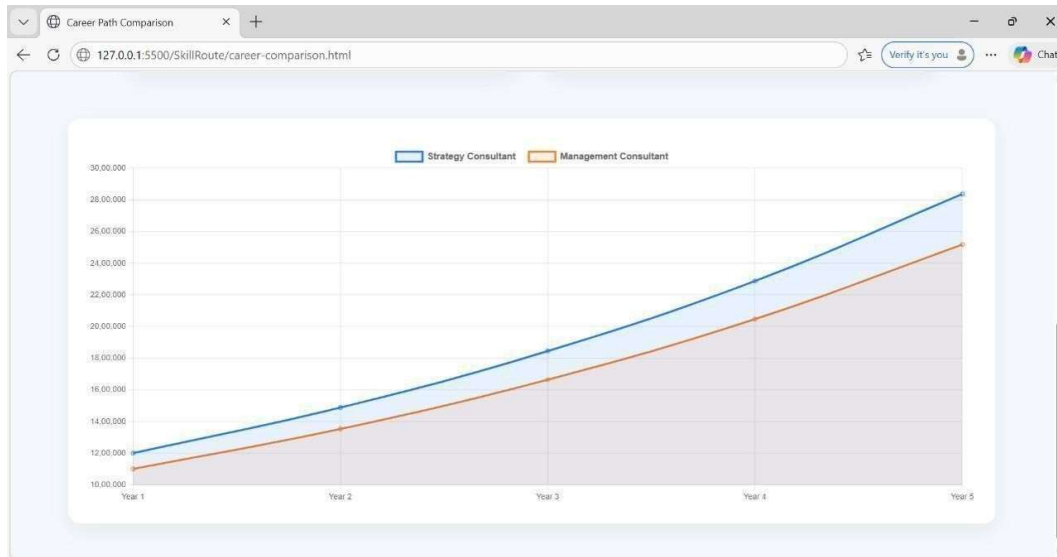
Screenshot 2 Registration page



Screenshot 3 Home page



Screenshot 4 Roadmap Generation



Screenshot 5 Career Comparison Page

### Conclusion

Skill Career Explorer provides an intelligent solution for career guidance by analyzing user skills and recommending suitable career paths. The system improves employability by identifying missing competencies and generating structured learning roadmaps. It also supports decision-making through career comparison and progress tracking features. In the future, the system can be enhanced by integrating job portals, analyzing real-time industry trends, incorporating advanced AI-based career prediction, developing a mobile application, adding gamification features, and providing mentorship support.

### Sustainable Development Goals

The Skill Career Explorer contributes to Sustainable Development Goals by promoting quality education, supporting innovation, and reducing inequalities. The system improves access to career guidance, enhances skill development, and provides equal opportunities for individuals from diverse backgrounds.

### References

1. H. Sajid, J. Kanwal, S. U. R. Bhatti, "Resume Parsing Framework for E-Recruitment," Proceedings of the International Conference on Ubiquitous Information Management and Communication, 2022.
2. M. Harshada, A. N. Kshirsagar, N. Sonali, "NLP Based Resume Parser Analysis," Journal of Emerging Technologies and Innovative Research, vol. 10, no. 5, 2023.
3. E. Khan, H. Patel, A. Shaikh, "Resume Parser and Summarizer," International Journal of Advanced Research in Science, Communication and Technology, 2023.
4. K. Sinha, M. A. K. Akhtar, M. Kumar, "Break Down Resumes into Sections to Extract Data and Perform Text Analysis using Python," International Journal on Recent and Innovation Trends in Computing and Communication, 2023.
5. T. Mikolov, K. Chen, G. Corrado, J. Dean, "Efficient Estimation of Word Representations in Vector Space," arXiv preprint arXiv:1301.3781, 2013.
6. J. Devlin, M. Chang, K. Lee, K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," Proceedings of NAACL, 2019.
7. F. Sebastiani, "Machine Learning in Automated Text Categorization," ACM Computing Surveys, vol. 34, no. 1, pp. 1-47, 2002.
8. P. Lops, M. de Gemmis, G. Semeraro, "Content-based Recommender Systems: State of the Art and Trends," Recommender Systems Handbook, Springer, 2011.
9. G. Salton, C. Buckley, "Term-weighting Approaches in Automatic Text Retrieval," Information Processing & Management, 1988.
10. S. Russell, P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd Edition, Pearson, 2010.
11. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016.
12. J. Han, M. Kamber, J. Pei, *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2012.
13. M. Ricci, L. Rokach, B. Shapira, "Recommender Systems Handbook," Springer, 2015.
14. K. Paparrizos, M. Cambazoglu, A. Gionis, "Machine Learned Job Recommendation," Proceedings of RecSys Conference, 2011.
15. A. Jain, P. Sharma, "AI-Based Career Recommendation System Using Machine

Learning,” International Journal of Computer Applications, vol. 183, no. 22, 2021.