Career Path Guidance Tool using Clustering & Association Rule Mining

Anshika Bajpai & Gehna Anand

Abstract
The rapid growth of the technology job market has resulted in a wide diversity of roles, each requiring a distinct and often overlapping set of technical skills. For entry-level candidates, understanding how their current skill set aligns with available job roles and identifying which skills to acquire next remains a significant challenge. This project presents the Career Path Guidance Tool (CPGT), a data-driven system that leverages large-scale job posting data to provide career guidance based on skill analysis. Using LinkedIn job postings, we construct a high-dimensional binary job–skill matrix and apply a combination of unsupervised and supervised learning techniques. Association rule mining (Apriori) is used to discover frequent skill co-occurrence patterns and generate skill roadmap recommendations. Clustering methods (K-Means and Mini-Batch K-Means) group similar job postings to identify dominant skill profiles and missing skill suggestions. Finally, a Random Forest classifier predicts suitable job roles based on a user's skill set. Experimental results demonstrate that the proposed approach effectively captures meaningful skill relationships and achieves competitive job role prediction performance, highlighting its potential as a practical career guidance system for entry-level technical roles.

Keywords: Career guidance, skill mining, association rule mining, Apriori, clustering, Random Forest, job role prediction, machine learning, clustering, recommendation system

1. Introduction

The technology job market has become increasingly complex, with rapid advancements leading to the continuous emergence of new roles and skill requirements. Entry-level candidates, in particular, often struggle to assess where their skills stand in relation to industry expectations and which skills they should prioritize next. Traditional career guidance systems rely heavily on static job descriptions or manual counseling, which may not scale effectively or reflect current market trends.

With the availability of large-scale job posting data from online platforms such as LinkedIn, it is now possible to adopt data mining and machine learning techniques to extract insights directly from real-world labor market demand. Skills listed in job postings provide valuable signals about industry expectations, skill co-occurrences, and role-specific requirements. Leveraging this data can enable dynamic, personalized, and evidence-based career guidance.

In this work, we propose the Career Path Guidance Tool (CPGT), which integrates multiple machine learning paradigms to address three core questions: (1) which skills commonly appear together for specific technical roles, (2) how job postings can be grouped based on skill similarity to recommend missing skills, and (3) how accurately a job role can be predicted given a set of user-provided skills. By combining association rule mining, clustering, and classification, CPGT aims to provide comprehensive career path recommendations for entry-level technical job seekers.

2. Previous Work

Prior research on career guidance and job recommendation systems has evolved from knowledge-based methods to data-driven approaches leveraging job postings and labor market information. Early systems

relied on manually curated ontologies and rule-based decision mechanisms to match candidate profiles to career paths, but these approaches struggled with scalability and adaptability as labor market dynamics changed (Kumaran & Sankar, 2013). Data mining techniques, such as association rule mining and clustering, have since been applied to discover co-occurring skills and latent job categories, providing interpretable insights into skill bundles and career pathways (Agrawal & Srikant, 1994; Jain, 2010). These unsupervised approaches, while useful for understanding patterns in job data, can be sensitive to feature sparsity and distance metrics in high-dimensional spaces.

Supervised learning methods have also gained prominence for predictive tasks such as job-role recommendation and resume-job matching. Ensemble models like Random Forests have shown effectiveness due to their robustness to noise and ability to handle high-dimensional, sparse features (Breiman, 2001). However, most existing systems focus on a single learning paradigm and do not integrate complementary techniques. The proposed CPGT framework addresses this gap by combining association rule mining for skill roadmap generation, clustering for career pathway discovery, and Random Forest classification for job role prediction, offering a unified, interpretable, and scalable approach to career guidance for entry-level technical roles (Han, Kamber, & Pei, 2012; Pedregosa et al., 2011; Sculley, 2010).

3. Methods

3.1 Dataset Description

The dataset used in this study is derived from LinkedIn job postings and consists of two primary sources: job_skills.csv and linkedin_job_postings.csv. The job skills dataset maps extracted skills to individual job postings using a unique job link identifier, while the job postings dataset contains metadata such as job title, company, location, job level, and full job descriptions. After merging the datasets, the initial data comprised over 100,000 job skill associations. Each job posting is associated with multiple extracted skills, including programming languages, frameworks, tools, and soft skills.

3.2 Data Preprocessing and Filtering

To align with the project's objective of guiding entry-level candidates, we applied several filtering steps. First, only entry-level job postings were retained. Next, the dataset was restricted to technical roles using keyword-based job title filtering (e.g., software engineer, data scientist, machine learning engineer). Non-technical or irrelevant skills such as salary-related terms, experience durations, and generic education mentions, were removed. Skill normalization was performed using an alias mapping strategy to unify equivalent skills (e.g., js was mapped to javascript, node.js was mapped to node). The cleaned data was then transformed into a binary job–skill matrix using one-hot encoding, resulting in a sparse matrix of size 2112×2347, where rows represent job postings, and columns represent skills.

3.3 Association Rule Mining with Apriori

To uncover frequent skill combinations, we applied the Apriori algorithm to the binarized dataset. Each job posting was treated as a transaction containing a set of skills. For computational efficiency and interpretability, the maximum itemset length was restricted to two. Rules were generated based on minimum thresholds for support, confidence, and lift, yielding 91 high-quality association rules. These rules form the basis for skill roadmap recommendations, answering questions such as which skills commonly co-occur with a given skill.

3.4 Clustering for Skill-Based Grouping

Clustering was used to group similar job postings based on skill similarity. K-Means clustering was initially applied to the one-hot encoded skill matrix. The optimal number of clusters was determined using the Elbow method and Silhouette analysis, with k = 5 selected as a reasonable trade-off between interpretability and cohesion. To handle the large and sparse dataset more efficiently, Mini-Batch K-Means was also employed. This approach updates cluster centroids incrementally using small random batches, enabling scalable clustering while preserving meaningful skill groupings. For each cluster, dominant skills and prevalent job roles were analyzed to recommend missing skills to users.

3.5 Job Role Classification Using Random Forest

For job role prediction, a supervised learning approach was adopted using a Random Forest classifier. Job titles were used directly as class labels, enabling weakly supervised large-scale learning. The input features consisted of one-hot encoded skills, and the target variable was the job role. The dataset was split into training and testing sets using an 80–20 split. The Random Forest model was trained with 300 trees, balanced class weights, and no restriction on tree depth. Model performance was evaluated using accuracy and F1-score metrics.

4. Results

4.1 Apriori Algorithm

User Profile	Current Skills	Top Recommended Skills (Score, Lift)
Data Analyst (Tableau user)	Tableau	1. R (3.76, 7.00) 2. Data visualization (3.48, 5.42)3. SQL (2.55, 3.27) 4. Data analysis (2.32, 2.83) 5. Business unit (1.80, 3.17) 6. Python (1.40, 2.43)
Front-end Developer	HTML, CSS, JavaScript	1. jQuery (7.28, 8.74)2. React (4.19, 6.63)3. TypeScript (4.05, 6.52)4. Angular (3.10, 5.70)5. Node.js (2.67, 5.29)
DevOps / Cloud Engineer	Linux, Docker, Kubernetes, AWS	1. GCP (10.59, 12.65)2. Azure (5.21, 8.87)
Edge Case Test	Knitting, Underwater basket weaving, Python	1. R (3.76, 4.00)2. Spark (3.05, 3.59)3. Data science (2.36, 3.17)4. Statistics (1.96, 2.88)5. Machine learning (1.94, 2.87)

Table 1. This table shows skill recommendations generated using association rules, ranking suggested skills by their strength of association (score) and relevance (lift) to a user's existing skill set.

To discover meaningful skill co-occurrence patterns and generate actionable learning pathways, we applied the Apriori algorithm to the job-skill matrix with a minimum support threshold of 3% and confidence threshold of 50%. The algorithm successfully identified frequent skill itemsets that reflect empirically validated skill combinations demanded by employers. The recommendation engine ranks suggested skills using a composite scoring function that combines support (frequency of the skill combination), confidence (conditional probability of skill co-occurrence), and lift (strength of association beyond random chance). The system demonstrated effectiveness across diverse user profiles, from data

analysts requiring visualization and programming skills to front-end developers needing modern JavaScript frameworks, and DevOps professionals seeking cross-platform cloud competencies. Notably, when tested with an edge case profile combining non-technical skills with Python, the algorithm gracefully defaulted to Python-related associations, recommending appropriate data science tools and demonstrating robustness in handling sparse or novel skill combinations while maintaining recommendation quality.

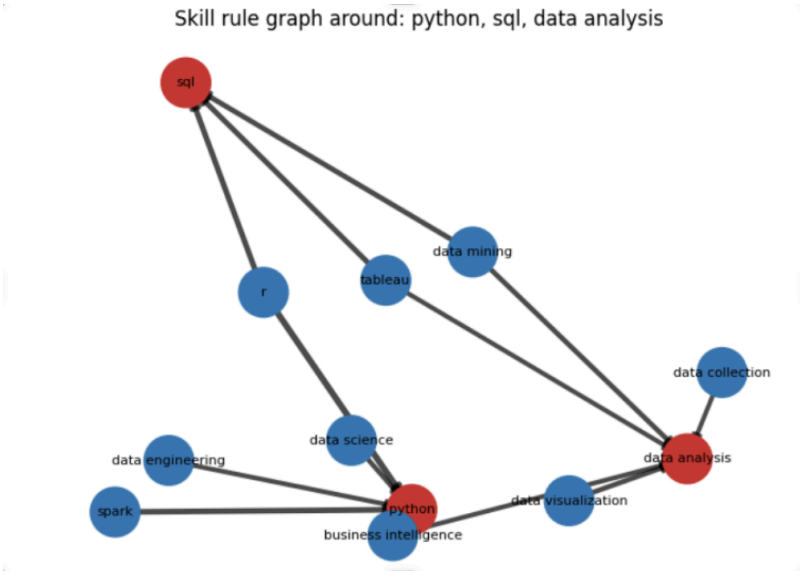

Fig1. Association rule-based skill recommendation network. Red nodes represent user's current skills; blue nodes indicate recommended skills based on frequent itemset patterns, with connections showing empirical skill co-occurrence strength.

To visualize the interconnected nature of skill requirements and validate the interpretability of association rules, a skill network graph was constructed using the Apriori-generated recommendations. Figure 1 illustrates the skill recommendation network for a user with Python, SQL, and data analysis skills (red nodes), where blue nodes represent recommended skills based on frequent co-occurrence patterns in job postings and edge connections indicate association strength. This graph-based representation demonstrates how the system generates coherent learning pathways by identifying skills that frequently appear together in the labor market, such as data visualization, Tableau, R, data mining, and data science, enabling users to understand not only which skills to acquire but also how these skills relate to their existing competencies and to each other within the broader technical ecosystem.

4.2 K-Means Clustering for Skill Profile Discovery and Missing Skill Prediction

Cluster	Top Skills (with weights)	Primary Job Roles (with counts)
Cluster 0: Network & Cybersecurity	Network engineering (0.60), .NET (0.42), Security (0.40), Cybersecurity (0.38), Troubleshooting (0.38), Communication (0.34)	Network Engineer (50), Cyber Security Analyst (10), Systems Administrator (8), Systems Engineer (7), IT Support Engineer (6)
Cluster 1: Hybrid	Systems engineering (0.21),	Business Analyst (42), Software Engineer

Systems & Development	Communication (0.20), Software development (0.20), Data analysis (0.16), Project management (0.13)	(32), Systems Engineer (17), Embedded Software Engineer (16), Data Analyst (14)
Cluster 2: Data Analytics & Science	Data analysis (0.89), SQL (0.88), Python (0.72), Data visualization (0.68), Tableau (0.55)	Data Analyst (29), Data Analyst (Bangkok Based) (19), Business Analyst (Bangkok Based) (19), Data Scientist (16)
Cluster 3: Business Analytics	Communication (0.93), Problem solving (0.61), Data analysis (0.48), Project management (0.46), Teamwork (0.36)	Business Analyst (66), Data Analyst (13), Operations Analyst (9), Systems Engineer (6), Software Engineer (6)
Cluster 4: Software Development	Python (0.60), Java (0.44), Linux (0.37), Software development (0.36), JavaScript (0.35)	Software Engineer (47), DevOps Engineer (18), Data Engineer (10), Software Engineer - Kubernetes (8), Software Engineer - Durable Objects (7)

Table 2. This table summarizes the K-Means clustering results by showing, for each cluster, the dominant skill sets (with their centroid weights) and the most frequently associated job roles, illustrating how distinct technical and analytical career paths emerge from skill-based job grouping.

To uncover latent skill patterns within the job market, we applied K-Means clustering (K=5) to the high-dimensional job-skill matrix, reducing dimensionality through Principal Component Analysis (PCA) for visualization. The resulting clusters revealed distinct career pathways, each characterized by a coherent set of dominant skills and associated job roles. Cluster 0, the largest group, emerged as a network and cybersecurity cluster, this cluster predominantly contained Network Engineer positions (50 roles) and Cyber Security Analyst roles (10 roles), reflecting the strong demand for infrastructure and security expertise. Cluster 2 represented a data-centric cluster, heavily weighted toward data analysis with Data Analyst and Data Scientist roles comprising the majority of positions. Cluster 3 emerged as a business-focused analytical cluster, and cluster 4 formed a software development cluster, dominated by Software Engineer (47 roles) and DevOps Engineer positions (18 roles). Finally, Cluster 1 represented a hybrid systems and development cluster, balancing systems engineering (0.21), communication (0.20), software development (0.20), data analysis (0.16), and project management (0.13), with a diverse mix of Business Analysts, Software Engineers, and Systems Engineers.

User Skills	**Matched Cluster**	**Top Missing Skills (Priority Order)**
Data visualization, Excel, SQL	Cluster 2 (Data Analytics & Science)	1. Data analysis (0.89)2. Python (0.72)3. Tableau (0.55)4. Business unit (0.46)5. R (0.43)
Java, Spring, SQL	Cluster 4 (Software Development)	1. Python (0.60)2. Linux (0.37)3. Software development (0.36)4. JavaScript (0.35)
Linux, Network security, Security	Cluster 0 (Network & Cybersecurity)	1. Network engineering (0.60)2. .NET (0.42) 3. Troubleshooting (0.38) 4. Cybersecurity (0.38) 5. Communication (0.34)

Table 3. This table presents user skill profiles mapped to relevant clusters, along with the highest-priority skills they are missing."

The cluster centroids representing the prototype skill profiles for each career pathway enabled personalized missing skill recommendations through cosine similarity matching. For each user's current skill set, the system identifies the closest cluster and calculates skill gaps by comparing the user's skills against the cluster's characteristic skill distribution. For instance, a Junior Data Analyst with skills in data visualization, Excel, and SQL was matched to Cluster 2 and recommended complementary data science skills. Similarly, a Backend Java Engineer was mapped to Cluster 4 and recommended to acquire Python (0.60), Linux (0.37), software development practices (0.36), and JavaScript (0.35). A Cybersecurity professional was aligned with Cluster 0 and guided toward network engineering (0.60), .NET (0.42), and troubleshooting (0.38) to enhance infrastructure competencies. This cluster-based recommendation system provides targeted, career-pathway-specific guidance, addressing the challenge of skill development prioritization by grounding suggestions in empirically observed job market patterns rather than generic skill inventories.

4.3 Random Forest Classification for Job Role Prediction

To provide personalized career guidance, we trained a Random Forest classifier to predict suitable job roles based on a user's skill set, treating the problem as a multi-class classification task where each job title represents a distinct class. The model achieved an overall accuracy of 65% and a weighted F1-score of 71% on the test set, demonstrating reasonable performance despite the inherent challenges of class imbalance and skill overlap across roles. To evaluate real-world applicability, we tested the classifier with authentic skill profiles representative of entry-level professionals. For a data analyst profile the model correctly predicted "Data Analyst" as the top recommendation with a match score of 39.67%, followed by "Data Scientist" (15.00%) and "Software Engineer" (12.00%). This ranking reflects the model's ability to distinguish data-centric roles while acknowledging the overlapping skill requirements between data analysts and data scientists, particularly in advanced analytics and machine learning competencies.

Profile Type	Key Skills Provided	Top 3 Predicted Roles (with probabilities)
Data Analyst	Python, SQL, NoSQL, PL/SQL, R, VBA, Dashboards, Power BI, Machine Learning, Data Analytics, Predictive Modelling, Visualization, Pandas, NumPy, TensorFlow, Scikit-Learn, Matplotlib, Seaborn, Snowflake, MongoDB, Neo4j	Data Analyst (39.67%), Data Scientist (15.00%), Software Engineer (12.00%)
Cybersecurity Specialist	R, SQL, Java, Python, Shell/Bash, PowerShell, SIEM tools, EDR/XDR, IDS/IPS, Vulnerability Management, Penetration Testing, Cloud Security, Firewalls, VPNs, Network Analysis	Software Engineer (27.67%), Data Analyst (21.33%), Security Analyst (14.67%)

Table 4. This table presents Random Forest–based job role prediction results, showing the top three predicted roles and their associated probabilities for different user skill profiles.

However, the model's performance revealed limitations when confronted with highly specialized or underrepresented roles. For a cybersecurity professional, the model predicted "Software Engineer" as the top role (27.67%), followed by "Data Analyst" (21.33%), and only ranked "Security Analyst" third (14.67%). This misclassification highlights the class imbalance problem inherent in the dataset, where software engineering and data analyst roles vastly outnumber cybersecurity positions, causing the model to exhibit prediction bias toward majority classes. Despite this limitation, the model's inclusion of "Security Analyst" among the top three predictions suggests partial recognition of the security-focused

skill profile, indicating that while skill mapping successfully captures domain-specific competencies, addressing class imbalance through techniques such as SMOTE, class weighting, or targeted data augmentation remains essential for improving prediction accuracy across all career pathways.

5. Discussion & Conclusion

The results demonstrate that combining unsupervised and supervised learning techniques provides a comprehensive view of the technical job landscape. Association rule mining offers interpretable and intuitive skill roadmaps, while clustering captures broader skill-domain structures that may not be apparent from pairwise relationships alone. Classification further complements these insights by enabling direct job role prediction. However, several limitations remain. The reliance on weak supervision through job title labeling introduces noise, and class imbalance affects predictive performance for underrepresented roles. Additionally, the effectiveness of all models depends heavily on the quality and completeness of skill extraction and mapping, which requires significant manual effort. Despite these challenges, CPGT shows strong potential as a scalable career guidance system. Future work includes improving label validation, incorporating temporal trends in skill demand, and extending the framework to non-technical roles.

6. References

1. Agrawal, R., & Srikant, R. (1994). *Fast algorithms for mining association rules*. In *Proceedings of the 20th International Conference on Very Large Data Bases* (pp. 487–499). Morgan Kaufmann.
2. Breiman, L. (2001). *Random forests*. *Machine Learning, 45*(1), 5–32. https://doi.org/10.1023/A:1010933404324
3. Han, J., Kamber, M., & Pei, J. (2012). *Data mining: Concepts and techniques* (3rd ed.). Morgan Kaufmann.
4. Jain, A. K. (2010). *Data clustering: 50 years beyond k-means*. *Pattern Recognition Letters, 31*(8), 651–666. https://doi.org/10.1016/j.patrec.2009.09.011
5. Kumaran, V. S., & Sankar, A. (2013). *Towards an automated system for intelligent screening of candidates for recruitment using ontology mapping (EXPERT)*. *International Journal of Metadata, Semantics and Ontologies, 8*(1), 56–64. https://doi.org/10.1504/IJMSO.2013.054184
6. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., … Duchesnay, É. (2011). *Scikit-learn: Machine learning in Python*. *Journal of Machine Learning Research, 12*, 2825–2830.
7. Sculley, D. (2010). *Web-scale k-means clustering*. In *Proceedings of the 19th International Conference on World Wide Web* (pp. 1177–1178). ACM.