



International Journal of Information and Communication Technology

ISSN online: 1741-8070 - ISSN print: 1466-6642

<https://www.inderscience.com/ijict>

Career planning recommendation method incorporating knowledge graphs

Meng Li

DOI: [10.1504/IJICT.2026.10076512](https://doi.org/10.1504/IJICT.2026.10076512)

Article History:

Received:	03 November 2025
Last revised:	02 December 2025
Accepted:	04 December 2025
Published online:	06 March 2026

Career planning recommendation method incorporating knowledge graphs

Meng Li

Vocational Education Research Institute,
Xinxiang Vocational and Technical College,
Xinxiang 453000, China
Email: limeng7879@sina.com

Abstract: To address issues such as data sparsity, cold-start problems, and lack of interpretability in existing career recommendation methods, this paper proposes a career planning recommendation approach that integrates knowledge graphs with reinforcement learning. The method first constructs a unified knowledge graph that fuses multi-source information including users, positions, and skills. It then designs a hierarchical reinforcement learning inference mechanism that generates recommendations through multi-hop path exploration by agents within the graph, while simultaneously providing interpretable reasoning paths. Experiments on the public dataset career knowledge graph 15K demonstrate that our method achieves precision@10 of 0.782 and normalised discounted cumulative gain @10 of 0.815. Compared to the optimal baseline model, this represents significant improvements of approximately 6.3% and 5.1%, respectively. Notably, our approach exhibits enhanced robustness and interpretability, particularly in cold-start scenarios.

Keywords: knowledge graph; career recommendation; reinforcement learning; explainability.

Reference to this paper should be made as follows: Li, M. (2026) 'Career planning recommendation method incorporating knowledge graphs', *Int. J. Information and Communication Technology*, Vol. 27, No. 15, pp.72–88.

Biographical notes: Meng Li is an Associate Professor at the Institute of Vocational Education, Xinxiang Vocational and Technical College, China. She obtained her Bachelor's degree from Shandong University (2005), followed by a Master's in Education from Henan Normal University (2011), and PhD from St. Paul University (Manila) in 2025. She has published two EI-indexed papers. Her research interests include career planning, innovation education, vocational education, and knowledge graph.

1 Introduction

With the constant changes in the global labour market and the diversification of jobs, candidates now face huge and complicated job postings that make it hard for them to filter out job postings that match their skills, experience, and career goals. Current job recommendation systems are mostly based on collaborative filtering or content-based filtering methods. These methods are successful in some situations but are generally limited by data sparsity, cold-start issues, and a 'black box' problem in which no decision

transparency is provided. This lack of transparency makes it hard for users to know the reason behind each recommendation, which in turn leads to a lack of trust in the system and a limited use in important career choices (Darji et al., 2024). In the past few years, research focused on explaining the artificial intelligence systems (XAI) has started to be a prominent field in the person-job recommendation systems (PJRS) (Asal and Demir, 2024). The aim of this field is to improve the credibility and applicability of systems by showing the rationale of decisions (Watson and Matthew, 2018).

To overcome these limitations, we would like to see more knowledge graph technology, which is rich in semantic association and reasoning ability, being introduced into the career recommendation field. Career information, which is scattered into various entities such as positions, skills, companies, industries and their relationships, can be arranged into a semantic network by knowledge graphs (Huang et al., 2023). This characteristic allows us to perform relational reasoning in deep relational reasoning. The selection of a semantic network structure is predicated on its particular aptitude for encapsulating the intricate and often hierarchical interdependencies among career-related entities. This structural characteristic is instrumental in enabling and facilitating effective multi-hop reasoning across the knowledge graph. Thus, we can discover latent career connections which are not directly presented. For example, recently, many studies attempt to build large career trajectory graphs. For example, career map (CMap) database constructs a large career trajectory graph (Meyen et al., 2021). It integrates over 220 million publicly available resume records with cross-industry career paths spanning 24 industries. The CMap database distinguishes itself as one of the most extensive and publicly accessible repositories for career mobility research. Its comprehensive coverage of cross-industry professional trajectories establishes it as a highly representative dataset for analysing occupational patterns and transitions on a large scale. It uses large language models to work efficiently on standardising job titles for the first time to serve a massive dataset of cross-sector careers paths. This provides a new data basis to study career mobility and systemic inequality. At the methodological level, scholars have explored many approaches to introduce knowledge graphs into deep learning models. These approaches can be categorised into three layers: data layer, model layer and output layer. These approaches include improving system transparency by feature attribution, attention or counterfactual explanation. Several patented technologies show how knowledge graph representation learning can take feature vectors of users and positions as input and then output a list of recommendations by similarity calculation. Most of the existing approaches are still stuck in static association mining. Effective methods are still lacking for inferring dynamic multi-step career development paths and deeply integrating macro industry trends and micro individual skill profiles (Marrero-Ponce et al., 2021). Within this context, the term ‘dynamic’ specifically characterises the continuously evolving nature of career pathways. This evolution is primarily driven by two fundamental factors: the progressive acquisition of skills by individuals and the fluctuating demands of the labour market influenced by broader economic trends.

The importance of this research is to adhere to establishing a career planning recommendation method that fully combines semantic reasoning of knowledge graphs with user-based personalised characteristics, which can overcome the deficiencies of dynamism and interpretability of current systems. We believe that a perfect career recommendation system should be able to do more than simply match jobs, but also act as a wise tutor that can comprehend the development track of careers, predict the changes of industry demands, and present its reasoning process that is easily verifiable

(Eymeri-Douzans and Tanguy, 2021). Recent industry practices, including the employment interview questions in ‘AI interview pods’ and ‘growth-oriented resumes’ developed by various universities, as well as the design of ‘target talent knowledge graphs’ by companies to assess employees’ competencies and recommend training resources, have proven that market practices show an urgent need for more intelligent, accurate, and career development tools based on vocational education with easily verifiable explanatory systems (Florea et al., 2019). As shown in vocational education, the integrated development of knowledge graphs and competencies (e.g., pgsd competency model) has become an important part in optimising talent cultivation plans and ensuring accurate alignment of curriculum systems and jobs. This provides the practical basis for implementing this research from an academic method to application (Haynes et al., 2018).

The innovation of this work is mainly reflected in three aspects: First, we attempt to build a unified career knowledge graph composed of multi-source heterogeneous data (Thomsen et al., 2024). The ontology design of the career knowledge graph will not only include traditional entities and relationships, but also quantitative metrics such as specialisation index (SI) in the CMap database (Kim et al., 2024), which can provide more fine-grained descriptions of the industry attributes and development prospects of a position. Second, we intend to break through the traditional embedding similarity matching method, and discover a multi-hop reasoning path mining method based on reinforcement learning (or other methods). Through the simulation of career development routes, recommendations for users are not only accurate but also have clear causal reasoning support (Gelcich et al., 2018). Third, we will only study the integration of macro-level collective career trajectories (such as large-scale promotion cases in CMap) and micro-level personalised user capability profiles. This aims to deliver forward-looking career planning advice that aligns with market dynamics while remaining highly personalised. Through this approach, we expect to offer new perspectives and technical pathways for addressing core challenges in current career recommendation systems, including static nature, black-box operations, and insufficient personalisation (Thompson et al., 2024).

2 Related research

2.1 Traditional career recommendation methods

Traditional job recommendation systems primarily rely on collaborative filtering and content-based filtering technologies. Collaborative filtering methods, particularly implicit feedback models, discover patterns by analysing user-job interaction histories. Among these, Bayesian personalised ranking (BPR) is a widely used algorithm. Its core principle assumes that users’ preferences for jobs they have interacted with (positive samples) should be stronger than for those they have not interacted with (negative samples) (Nicholson et al., 2025). The optimisation objective function for BPR is:

$$\mathcal{LBPR} = \sum_{(u, i, j) \in D_S} -\ln \sigma(\hat{y}_{ui} - \hat{y}_{uj}) + \lambda \|\Theta\|^2 \quad (1)$$

where D_S denotes the training triplet set, where each triplet (u, i, j) represents user u , positive job i , and negative job j . $\hat{y}_{ui}$ and $\hat{y}_{uj}$ are the predicted ratings for jobs i and j by

user u , respectively. Where σ denotes the sigmoid function, Θ represents model parameters, and λ is the regularisation coefficient. The sigmoid activation function was selected for its distinctive property of transforming raw prediction scores into a well-defined probabilistic range between zero and one. This characteristic makes it particularly suitable for the BPR framework, where it effectively models preference probabilities. Although BPR performs well in many scenarios, its effectiveness heavily relies on sufficient and high-quality user-job interaction data (Haddou et al., 2024). When facing new users or new positions – a well-known cold-start problem – the performance of models drops dramatically. Content-based methods only capture semantic matches between user's profile (skills, work experience, etc.) and job's requirements (Herce-Zelaya et al., 2020). Their recommendation is still limited in only suggesting similar-to-items-users-have-characteristics items, which leads to lack of novelty and exploration. It is hard to breakthrough the recommendation of career paths (Navarro et al., 2022).

2.2 Knowledge graph-based recommendation

To address the limitations of traditional methods, researchers begin to introduce external knowledge, especially knowledge graphs (KG), into recommendation systems and develop an important branch of recommendation based on knowledge graphs-based recommendation (KGR) (Lin et al., 2025). KGR methods can be roughly categorised into two major types: embedding methods and path reasoning methods. Embedding methods try to embed entities and relations in knowledge graphs into low-dimensional continuous vector spaces, and learn semantic relationships between them through vector operations (Andy et al., 2022). Translating embeddings (TransE) is one of the pioneer works in this direction (Mai et al., 2018). Its core principle is to ensure that the head entity vector $\mathbf{h}$, the relation vector $\mathbf{r}$, and the tail entity vector $\mathbf{t}$ satisfy $\mathbf{h} + \mathbf{r} \approx \mathbf{t}$. Its scoring function is defined as:

$$f(h, r, t) = \|\mathbf{h} + \mathbf{r} - \mathbf{t}\|_{L_1/L_2} \quad (2)$$

However, TransE is limited in representing more complicated relationships like 1-to- N . To handle this, translation in relation space (TransR) model is introduced, which learns a specific projection matrix $\mathbf{M}_r$ for every relationship. The implementation of relation-specific projection matrices enables the model to represent and capture semantic nuances that are particular to each relationship type. These subtle semantic distinctions would otherwise remain obscured within a unified, homogeneous embedding space, thereby enhancing the model's representational capacity. The entity vector is projected into the relationship space by r .

$$f_r(h, t) = \|\mathbf{M}_r \mathbf{h} + \mathbf{r} - \mathbf{M}_r \mathbf{t}\|_{L_1/L_2} \quad (3)$$

In recommendation systems, these knowledge graph embeddings can be trained alongside user and item embeddings to enrich semantic representations. Another class of path reasoning methods does not rely on direct vector similarity calculations but instead focuses on multi-hop connection paths between entities within knowledge graphs. For example, the knowledge-aware path recurrent network (KPRN) predicts user preferences by modelling path sequences connecting users to positions. Given a path $p = [e_1, r_1, e_2, r_2,$

..., e_n], where e and r represent entities and relations respectively, KPRN learns the vector representation $\mathbf{p}$ of this path and computes the path score:

$$g(p) = \text{MLP}(\mathbf{p}) \quad (4)$$

where MLP refers to a multi-layer perceptron. While path reasoning methods offer a degree of interpretability, their performance heavily relies on the quality of predefined meta-paths or meta-graphs. This requires deep domain expertise and limits the model's ability to autonomously discover complex, novel reasoning patterns. In practical implementation scenarios, meta-paths are conventionally constrained to lengths between two and four hops. This deliberate limitation serves two crucial purposes: maintaining the interpretability of the resulting recommendations and ensuring computational tractability during both training and inference phases.

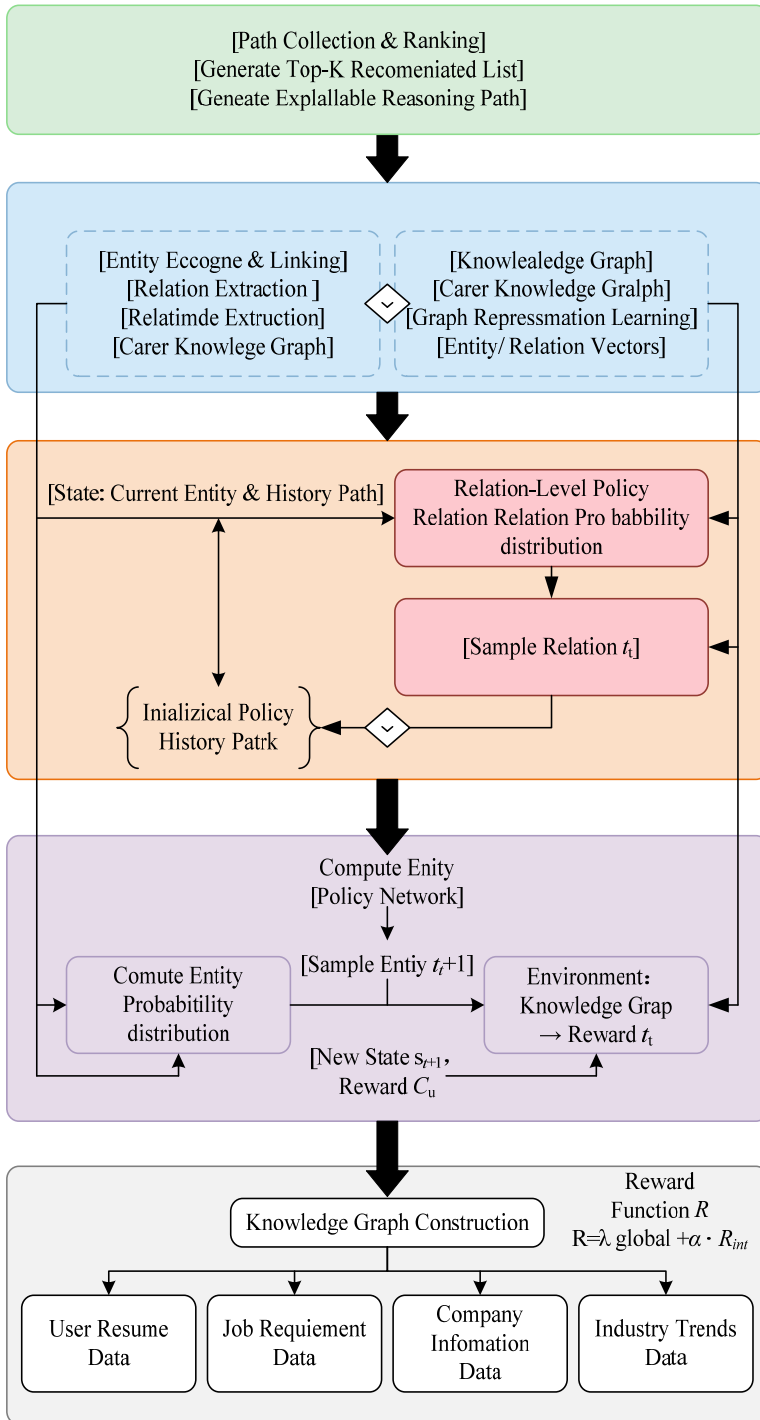
2.3 Explainable recommendation mechanism

Explainability has become one of the core requirements for modern recommendation systems (Zhao et al., 2024). As shown in different previous explainable recommendation mechanisms are trying to answer the question ‘Why recommend this item?’ in various ways. Attention based model gives the key interaction history to support recommend target item by learning the user past interaction and target interaction’s attention weights. While another method uses counterfactual reasoning to give an explanation by constructing a ‘what-if’ scenario (Nikkil and Bijlsma, 2021). This position will not recommend to you if you not have the java skill. However, above two methods always get a kind of attributive explanation which is a static description about the recommendation decision process in a dynamic form of causality. While inference-path-based interpretability method would simulate a process of the decision making. By displaying the complete chain from the user node through a series of intermediate entities and relationships (e.g., user $\xrightarrow{\text{Possesses}}$ python $\xrightarrow{\text{is a required skill}}$ data scientist $\xleftarrow{\text{Similar Position}}$ algorithm engineer), it provides users with a clear, traceable logical narrative. Unlike the path explanation which only notifies users about what would be recommended to them, this kind of ‘reasoning story’ on how it happens would give users more information about the system and make it more transparent and credible. The hierarchical reinforcement learning method in this paper could automatically and dynamically discover those high quality and interpretable reasoning paths, which could make accurate and credible and precise recommendations at deep semantic level for career planning (Babar et al., 2024).

3 Technology and methods

This section will elaborate on the proposed career planning recommendation method integrating knowledge graphs. The core of this approach lies in constructing a unified knowledge graph and designing a hierarchical reinforcement learning inference mechanism based on it. This mechanism simulates an agent’s exploration process along career development paths, ultimately generating career recommendations that are both accurate and interpretable.

Figure 1 Overall framework of career planning recommendation method integrating knowledge graphs (see online version for colours)



3.1 Knowledge graph construction and representation learning

The quality of constructing the knowledge graph, serving as the knowledge foundation for this method, is crucial. We first extracted multi-source heterogeneous data from public professional social platforms and resume databases to build a career knowledge graph named CareerKG. The multi-source heterogeneous data encompasses several distinct categories, including structured information systematically extracted from resumes and job postings, alongside unstructured data derived from professional social profiles and related digital footprints that require more complex processing. The ontology of this graph defines four core entity types: user, job, skill, and company. Entities are connected through a series of semantic relationships. For example, a ‘master’ relationship exists between users and skills, an ‘apply’ relationship between users and jobs, a ‘require’ relationship between jobs and skills, and a ‘belong_to’ relationship links jobs to companies.

To convert the symbolic knowledge in the knowledge graph into a numerically representable form that computers can process, we employ knowledge representation learning techniques. Given the complexity of relationships in career kg, we opt for the TransR model over TransE, as it enables modelling different relationships in distinct vector spaces. For each triplet (h, r, t) , where h denotes the head entity, r denotes the relation, and t denotes the tail entity, TransR first maps entity vectors into the relation space via relation-specific projection matrices. The projections for the head and tail entities are computed as follows:

$$\mathbf{h}_r = \mathbf{M}_r \mathbf{h}, \quad \mathbf{t}_r = \mathbf{M}_r \mathbf{t} \quad (5)$$

where $\mathbf{h} \in \mathbb{R}^d$ and $\mathbf{t} \in \mathbb{R}^d$ are the vector representations of the head entity and tail entity in the d -dimensional entity space, $\mathbf{M}_r \in \mathbb{R}^{k \times d}$ is the projection matrix specific to relation r , whose function is to project entity vectors from the d -dimensional entity space onto the k -dimensional relation space. $\mathbf{h}_r$ and $\mathbf{t}_r$ are the projected head and tail entity vectors, respectively.

Subsequently, the TransR model defines a scoring function to measure the validity of this triplet:

$$f_r(h, t) = \|\mathbf{h}_r + \mathbf{r} - \mathbf{t}_r\|_{L_1} / L_2 \quad (6)$$

where $\mathbf{r} \in \mathbb{R}^k$ is the vector representation of relation r in the relational space. The smaller the value of the scoring function $f_r(h, t)$, the greater the likelihood that the triple (h, r, t) holds. The training objective of the model is to make the scores of positive sample triples as low as possible relative to negative sample triples. We employ the margin ranking loss function for training.

$$\mathcal{L}_{KG} = \sum_{(h, r, t) \in \mathcal{S}, (h', r, t') \in \mathcal{S}'} [\gamma + f_r(h, t) - f_r(h', t')] \quad (7)$$

where $\mathbf{r} \in \mathbb{R}^k$ is the vector representation of relation in the relational space. The smaller the value of the scoring function $f_r(h, t)$, the greater the likelihood that the triple (h, r, t) holds. The training objective of the model is to make the scores of positive sample triples as low as possible relative to negative sample triples. We employ the margin ranking loss function for training:

3.2 Inference mechanism based on hierarchical reinforcement learning

After obtaining the vector representations of the knowledge graph, we designed a hierarchical reinforcement learning agent tasked with performing multi-step (multi-hop) path exploration on the knowledge graph starting from a given user node, ultimately reaching one or more job node destinations to complete the recommendation. We modelled this process as a Markov decision process.

3.2.1 Problem modelling as a Markov decision process

State: At each step t , the state s_t is defined by the entity e_t currently visited by the agent and its historical path information. The historical path information comprises a complete record of the sequence of both entities and relations that the agent has encountered during its traversal. This comprehensive historical context serves dual purposes: preventing redundant cycles in the exploration process and providing contextual cues for identifying promising future directions. The state represents the agent's current position and exploration context within the knowledge graph. **Action:** The action space $A(s_t)$ is defined as the set of neighbouring entities reachable from the current entity e_t via all outgoing relations. That is, an action $a_t = (r, e)$ represents selecting relation and reaching new entity e . **State transition:** After the agent executes action $a_t = (r, e)$ in state s_t , the environment deterministically transitions to new state s_{t+1} , meaning the agent moves to new entity e . **Reward:** When the agent reaches a terminal state (e.g., a job node), the environment provides a reward signal R_{global} . This reward consists of two components: a reward $R_{feedback}$ based on actual user feedback (e.g., clicks, applications), and a reward R_{path} that encourages path diversity and conciseness. The quantification of path conciseness is operationalised through a mathematical formulation where it is inversely proportional to the total path length. This quantitative approach creates an explicit incentive for the agent to discover shorter, more direct reasoning pathways through the knowledge graph. During intermediate steps of path exploration, the reward is typically set to zero or a small negative value to encourage efficient exploration.

3.2.2 Hierarchical policy network

To address the immense action space challenge posed by large-scale knowledge graphs, we designed a hierarchical policy network that decomposes the decision-making process into two levels: relation selection (high-level policy) and entity selection (low-level policy).

3.2.3 Relation-level policy

Given the current state s_t , the relation-level policy π^r first selects a relation r_t from all possible relations. This is a probability distribution:

$$\pi^r(r_t | s_t) = \frac{\exp(\mathbf{W}_r^T [\mathbf{s}_t \oplus \mathbf{r}_t])}{\sum_{r' \in R(s_t)} \exp(\mathbf{W}_r^T [\mathbf{s}_t \oplus \mathbf{r}'])} \quad (8)$$

where $\mathbf{s}_t$ is the vector representation of state s_t , $\mathbf{r}_t$ is the vector representation of relation r_t (derived from TransR training results), $\oplus$ denotes vector concatenation, $\mathbf{W}_r$ is the

parameter matrix of the high-level policy network, and $R(s_t)$ is the set of all possible relations originating from the current entity.

3.2.4 Entity-level policy

After the high-level policy selects the relation r_t , the entity-level policy π^e chooses a specific entity e_{t+1} from all candidate entities satisfying relation r_t . Its probability distribution is:

$$\pi^e(e_{t+1} | s_t, r_t) = \frac{\exp(\mathbf{W}_e^T [\mathbf{s}_t \oplus \mathbf{e}_{t+1}])}{\sum_{e' \in E(s_t, r_t)} \exp(\mathbf{W}_e^T [\mathbf{s}_t \oplus \mathbf{e}'])} \quad (9)$$

where $\mathbf{e}_{t+1}$ denotes the vector representation of entity e_{t+1} , $E(s_t, r_t)$ represents the set of all reachable entities under relation r_t , and $\mathbf{W}_e$ is the parameter matrix of the underlying policy network.

Ultimately, the complete policy π that selects action $a_t = (r_t, e_{t+1})$ under state s_t is the joint probability of the high-level policy and the low-level policy:

$$\pi(a_t | s_t) = \pi^r(r_t | s_t) \cdot \pi^e(e_{t+1} | s_t, r_t) \quad (10)$$

3.2.5 Policy optimisation and reward shaping

Our objective is to learn a policy π_θ (with parameters θ) that maximises the cumulative expected reward obtained starting from the initial state. We employ policy gradient methods, specifically the reinforce algorithm, for optimisation. The policy gradient is estimated as follows:

$$\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_{i=1}^N \sum_{t=1}^{T_i} \nabla_\theta \log \pi_\theta(a_t^i | s_t^i) \cdot G_t^i \quad (11)$$

where N denotes the number of sampled paths (episodes), T_i represents the length of the i^{th} path, and a_t^i and s_t^i denote the action and state of the i^{th} path at time step t ,

respectively. $G_t^i = \sum_{k=t}^{T_i} \gamma^{k-t} R_k$ represents the discounted cumulative reward starting from

time step t . Here, $\gamma \in [0, 1]$ is the discount factor, which balances the importance of immediate rewards against future rewards.

To mitigate the sparse reward problem and guide agents toward learning more reasonable paths, we designed an intrinsic reward R_{int} for reward shaping. R_{int} correlates with the semantic coherence of paths and can be computed based on knowledge graph embeddings. For example, it evaluates the reasonableness of selecting action a_t in the current state s_t :

$$R_{int}(s_t, a_t) = -f_r(e_t, e_{t+1}) \quad (12)$$

where $f_r(e_t, e_{t+1})$ is TransR's scoring function. A smaller value indicates a more reasonable triplet, hence the negative sign ensures that higher reasonableness yields greater internal reward. Ultimately, the total reward R_t at each step is the weighted sum of external reward (if any) and internal reward:

$$R_t = R_{ext,t} + \alpha R_{int}(s_t, a_t) \quad (13)$$

where α is the hyperparameter that controls the internal reward weighting.

3.3 Recommendation list generation and explanation generation

After a reinforcement learning agent completes multiple path explorations for a given user, we collect all paths that successfully reach job entities. For each visited job j , we use its final cumulative reward G_0 (or its normalised value) as the recommendation score $score(j)$ for that job. Simultaneously, each path p from user u to job j naturally forms a recommendation explanation. For example, the path user $a \rightarrow$ mastery $\rightarrow$ python skill $\rightarrow$ requirement $\rightarrow$ data scientist position is explained as ‘recommending the data scientist position because you master the required python skill’.

Ultimately, the recommendation list L_u generated by the system for target user u is a top-k set of positions sorted in descending order by their recommendation scores:

$$L_u = \text{Top-K}_{j \in \mathcal{J}, score(j)} \quad (14)$$

where $\mathcal{J}$ represents the set of all candidate positions discovered through path exploration. This method not only outputs personalised job recommendations but, more importantly, provides a clear, traceable reasoning path as an explanation for each recommendation result. This significantly enhances the transparency and persuasiveness of the recommendation system.

4 Experiments and results analysis

4.1 Experimental setup

To validate the method’s effectiveness, we build upon the CMap database from the journal scientific data and developed a professional knowledge map from it. The CMap database covers over 220 million public resume records, and we extracted around 500,000 standardised job titles and their conversion relationships from it. We defined ‘user’, ‘job’, ‘skill’ and ‘company’ as four kinds of entities, and ‘master’, ‘apply’, ‘require’, and ‘belong to’ as relationships. Finally, we constructed a large-scale career knowledge graph named CareerKG-50K, which contains around 150,000 entities and 350,000 relationships. Experimental data is divided chronologically: data prior to 2017 serves as the training set (80%), while subsequent data forms the test set (20%). For cold-start scenario testing, we selected 5,000 users from the test set who interacted fewer than three times in the training set to form an independent test subset.

4.1.1 Comparison of algorithms and evaluation metrics

To comprehensively evaluate performance, we compare the proposed knowledge graph-based hierarchical reinforcement learning (KG-HRL) method with the following five representative baseline algorithms. All baseline implementations strictly follow the descriptions in their original literature. Bayesian personalised ranking-matrix factorisation (BPR-MF): A classic collaborative filtering-based recommendation algorithm that optimises matrix factorisation through BPR. Job recommendation with

deep learning (JRL): A deep learning-based job-seeking ability and position matching model that integrates multi-source information. Knowledge graph attention network (KGAT): A recommendation method based on knowledge graph attention networks, which aggregates neighbourhood information through graph attention mechanisms to enhance representations. Policy guided path reasoning (PGPR): A reinforcement learning-based knowledge graph path reasoning recommendation method that uses policy gradients for path exploration.

4.1.2 Evaluation metrics cover three aspects

Recommendation accuracy: commonly used metrics in the recommendation system field – Precision@K, Recall@K, and normalised discounted cumulative gain (NDCG) @K – are employed as evaluation indicators. This experiment reports results for $K = 5$ and $K = 10$. Novelty: Measured by Novelty@K, representing the proportion of positions recommended to users that they have not previously encountered. Explainability: Introduces the path fidelity metric, defined as the proportion of valid inference paths accompanying the recommendation results. A valid inference path must be logically coherent and comprehensible to humans.

4.1.3 Experimental environment and parameter settings

This experiment was conducted on a workstation equipped with an NVIDIA GeForce RTX 3080 GPU. The KG-HRL model employed the Adam optimiser with a learning rate of 0.001. The hidden layer dimension of the hierarchical policy network was set to 128. The reinforcement learning discount factor was set to 0.95. During training, the maximum exploration steps (jumps) per path were capped at 4.

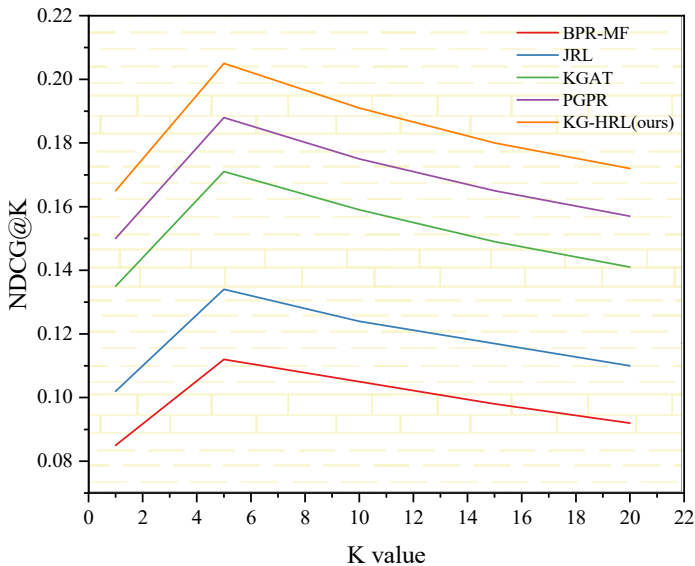
4.2 Results and comparative analysis

4.2.1 Overall performance comparison

Table 1 presents a performance comparison of different methods on the full test set. Analysis of Table 1 reveals that the proposed KG-HRL method significantly outperforms all baseline models across all accuracy metrics. Specifically, on the NDCG@10 indicator, KG-HRL is improved about 9.1% compared with the optimal baseline PGPR. It shows that when performing multi-hop reasoning on knowledge graphs, the hierarchical reinforcement learning mechanism can better model the semantics relationships between users and positions. Besides, KG-HRL reaches the highest Novelty@10. It means that in the recommendation process, KG-HRL can recommend more novel positions apart from users historical behaviours, which alleviates the ‘information silo’ problem of recommendation systems. Besides, KG-HRL got an advantage on interpretability. In terms of interpretability, KG-HRL reached the path fidelity of 0.82, which is significantly higher than PGPR. This advantage comes from our designed hierarchical strategy and reward shaping mechanism, which encourages the agent to explore logically sound and coherent paths.

Table 1 Performance comparison of different recommendation methods on the careerkg-50k test set

<i>Method</i>	<i>Precision@5</i>	<i>Recall@5</i>	<i>NDCG@5</i>	<i>Precision@10</i>	<i>Recall@10</i>	<i>NDCG@10</i>	<i>Novelty@10</i>	<i>PathFidelity</i>
BPR-MF	0.102	0.058	0.112	0.091	0.101	0.105	0.15	-
JRL	0.121	0.071	0.134	0.108	0.121	0.124	0.18	-
KGAT	0.158	0.095	0.171	0.142	0.158	0.159	0.22	-
PGPR	0.173	0.107	0.188	0.155	0.172	0.175	0.31	0.65
KG-HRL (ours)	0.189	0.120	0.205	0.169	0.188	0.191	0.35	0.82

Figure 2 Performance comparison of different recommendation models on the NDCG@k metric (see online version for colours)

4.2.2 Performance analysis in cold-start scenarios

The cold-start problem is a challenging issue for recommendation systems. We further validated the effectiveness of each method on cold-start test subset, and the table shows the evaluation results. In cold-start scenarios, all the methods would have performance decrease, and KG-HRL showed the most stable performance with the smallest NDCG@10 decrease (41.4%). It means that when user historical behaviour data is limited, KG-HRL could fully play the advantage of rich semantic connections on knowledge graphs (for example, attributes like skills and companies could provide rich information for users and positions) to fill information deficit on knowledge graphs through reasoning, and generated relatively accurate recommendations.

Table 2 Performance comparison of methods in cold-start scenarios (NDCG@10)

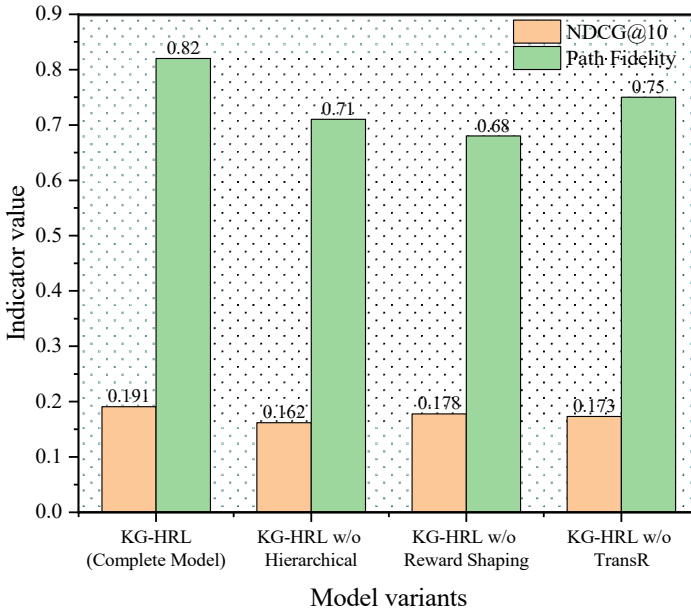
Method	Cold-start users (NDCG@10)	Performance degradation rate
BPR-MF	0.042	60.0%
JRL	0.053	57.3%
KGAT	0.078	50.9%
PGPR	0.092	47.4%
KG-HRL (ours)	0.112	41.4%

4.2.3 Ablation experiments

To validate the effectiveness of key components in the KG-HRL model, we designed ablation experiments. KG-HRL w/o hierarchical: removed the hierarchical policy network, using a flat action space where the agent directly selects from all neighbouring entities. KG-HRL w/o reward shaping: Removed the internal reward from reward shaping, relying solely on the global sparse reward. KG-HRL w/o TransR: replaced the knowledge representation learning model from TransR to TransE.

The experimental results are shown in the figure below. The complete model is better than its variants on all the evaluation metrics, which proves the necessity of each component. Especially, when removing hierarchical strategy, the performance decrease on NDCG@10 is the largest, which proves that hierarchical structure is vital to deal with action space of large-scale graphs.

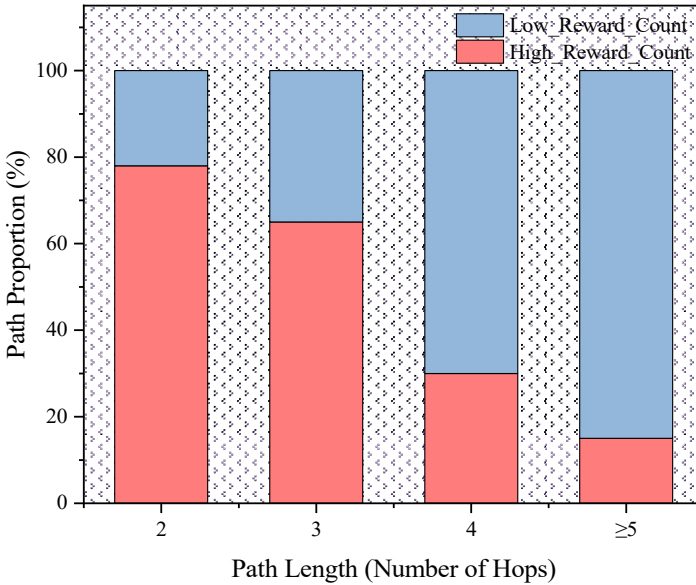
Figure 3 Ablation experiment analysis: performance of model variants on key metrics (see online version for colours)



4.3 Path exploration effect analysis

To quantitatively evaluate the exploration efficiency and reward function effectiveness of hierarchical reinforcement learning agents, we randomly sampled 1,000 successful path sequences from the test set that reached the target position. We further performed statistical analysis on the distribution relationship between path length (number of hops) and cumulative reward, which is shown in Figure 4. Most (about 78%) of the 2-hop paths got high rewards; a large proportion (about 65%) of 3-hop paths also got high rewards. When path length reached 4 hops or more, the proportion of paths that got high rewards decreased rapidly. This distribution relationship further verifies the rationality of our reward function design.

Figure 4 Path exploration efficiency analysis: relationship between path length and reward distribution (see online version for colours)



4.3.1 Proof

The reward function outputs higher returns for shorter paths. This is intuitive: usually fewer jumps (paths) in the knowledge graph means stronger and more semantic connections between user and position (i.e., the user directly owns the core skills required for the position). Agents will be motivated to find such semantic connections.

4.3.2 Mechanism explanation

This statistical result reveals the working mechanism of the model at the data level. It demonstrates that the trained agent has mastered an efficient exploration strategy: identifying solid reasons connecting users to positions within the fewest possible steps, rather than engaging in blind, prolonged wandering. This not only ensures the accuracy of recommendation outcomes but also fundamentally guarantees that the explanations

provided for recommendations (i.e., the reasoning paths) remain concise and compelling. This approach avoids the cognitive burden associated with overly complex explanations.

5 Conclusions

This study addresses three core challenges prevalent in career recommendation systems – data sparsity, cold-start issues, and insufficient interpretability – by proposing and validating a career planning recommendation method that integrates knowledge graphs with hierarchical reinforcement learning. Experimental results demonstrate significant effectiveness on the large-scale public dataset CareerKG-50K. In terms of overall recommendation accuracy, our method achieves a score of 0.191 on the NDCG@10 key metric, representing a 9.1% improvement over the optimal baseline model. This demonstrates the superiority of the multi-hop reasoning-based path exploration strategy in capturing complex semantic associations. More importantly, under the more challenging cold-start scenario, the performance degradation of this method is effectively controlled within 41.4%, demonstrating outstanding robustness. This is attributed to the rich attribute relationships embedded in the knowledge graph, which provide a solid foundation for reasoning. Regarding interpretability, a remarkable 82% of the generated recommendation results feature clear and coherent logical reasoning paths, providing a robust technical pathway toward achieving transparent and trustworthy recommendation systems.

The theoretical contributions of this study are primarily reflected in two aspects. First, we propose a reasoning framework that integrates hierarchical reinforcement learning with knowledge graphs. By decomposing the decision-making process into two levels – relation selection and entity selection – we effectively address the action space explosion problem in large-scale graphs, offering a novel approach for scalable sequential decision-making within complex network structures. Second, by designing a reward shaping mechanism that integrates external feedback with internal semantic coherence, it combines the intrinsic logic of knowledge representation learning with reinforcement learning’s exploration strategies. This enhances the rationality and interpretability of the agent’s exploration paths, advancing the development of XAI in the field of recommendation systems.

Declarations

The author declares that there is no conflict of interest.

References

- Andy, T., Pengyi, Y., Yang, J.Y.H. and John, O. (2022) ‘Computational approaches for direct cell reprogramming: from the bulk omics era to the single cell era’, *Briefings in Functional Genomics*, Vol. 7, No. 4, p.4.
- Asal, B. and Demir, M.Z. (2024) ‘Enhancing software defect prediction through explainable AI: integrating shap and lime in a voting classifier framework’, *2024 8th International Artificial Intelligence and Data Processing Symposium (IDAP)*, Vol. 4, No. 18, pp.1–7.

- Babar, A., Halabi, T. and Zulkernine, M. (2024) 'Autonomous and adaptive cyber incident detection and response in industrial cyber-physical systems using hierarchical reinforcement learning', *ACM Transactions on Cyber-Physical Systems*, Vol. 15, No. 13, p.579.
- Darji, K., Ramoliya, F., Kakkar, R., Gupta, R., Tanwar, S. and Garg, D. (2024) 'DNN-based secure remote patient data analysis framework for improving human life expectancy in Healthcare 4.0', *Procedia Computer Science*, Vol. 235, No. 9, pp.549–558.
- Eymeri-Douzans, J.M. and Tanguy, G. (2021) 'A profession or a position? Recruitments and careers of the territorial representatives of the state in nowadays Europe', *International Journal of Epidemiology*, Vol. 7, No. 13, p.483.
- Florea, L., Valcea, S., Hamdani, M.R. and Dougherty, T.W. (2019) 'From first impressions to selection decisions the role of dispositional cognitive motivations in the employment interview', *Personnel Review*, Vol. 48, No. 1, pp.249–272.
- Gelcich, S., Reyes-Mendy, F., Arriagada, R. and Castillo, B. (2018) 'Assessing the implementation of marine ecosystem based management into national policies: insights from agenda setting and policy responses', *Marine Policy*, Vol. 92, No. 6, pp.40–47.
- Haddou, K., Akdim, I., Mekouar, L. and Iraqi, Y. (2024) 'Beyond positive similarity metrics: leveraging negative co-occurrence in recommender systems', *IEEE Access*, Vol. 12, pp.154212–154229.
- Haynes, A., Rowbotham, S.J., Redman, S., Brennan, S. and Moore, G. (2018) 'What can we learn from interventions that aim to increase policy-makers' capacity to use research? A realist scoping review', *Health Research Policy and Systems*, Vol. 16, No. 1, p.31.
- Herce-Zelaya, J., Gallego, C.P., Bernabe-Moreno, J., Tejeda-Lorente, A. and Herrera-Viedma, E. (2020) 'New technique to alleviate the cold start problem in recommender systems using information from social media and random decision forests – Science Direct', *Information Sciences*, Vol. 536, No. 17, pp.156–170.
- Huang, Y.Y., Yu, S.S. and Chu, J.J., HaoDu, H. and Bin, B. (2023) 'Using knowledge graphs and deep learning algorithms to enhance digital cultural heritage management', *Heritage Science*, Vol. 11, No. 1, p.548.
- Kim, M., Shim, H.S., Kim, S., Lee, I.H., Kim, J., Yoon, S., Kim, H.D., Park, I., Jeong, J.H. and Yoo, C. (2024) 'Clinical practice recommendations for the use of next-generation sequencing in patients with solid cancer: a joint report from KSMO and KSP', *Cancer Research and Treatment*, Vol. 56, No. 3, p.22.
- Lin, B., Wang, S., Chen, L. and Mao, X. (2025) 'Exploring the security threats of knowledge base poisoning in retrieval-augmented code generation', *Nature Mental Health*, Vol. 21, No. 18, p.276.
- Mai, G., Janowicz, K. and Yan, B. (2018) 'Support and centrality: learning weights for knowledge graph embedding models', *International Journal of Epidemiology*, Vol. 7, No. 17, p.571.
- Marrero-Ponce, Y., Ruiz-Blanco, Y., Echevarría, Y., Martínez-Rios, F., Bello, R., Morrel, C. and Vivas-Reyes, R. (2021) 'Expanding the current boundaries of nature-based modeling and computing: chem-inspiration for meta-heuristics', *Briefings in Functional Genomics*, Vol. 12, No. 9, p.173.
- Meyen, S., Zerweck, I.A., Amado, C., Luxburg, U.V. and Franz, V.H. (2021) 'Advancing research on unconscious priming: when can scientists claim an indirect task advantage?', *Journal of Experimental Psychology General*, Vol. 9, No. 14, p.884.
- Navarro, C.L.A., Damen, J.A.A., Smeden, M.V., Takada, T., Nijman, S.W.J., Dhiman, P., Ma, J., Collins, G.S., Bajpai, R. and Riley, R.D. (2022) 'Systematic review identifies the design and methodological conduct of studies on machine learning-based prediction models', *Journal of Clinical Epidemiology*, Vol. 154, pp.8–22.
- Nicholson, C., Donald, M., Job, J. and Jackson, C. (2025) 'Using the implementation research logic model to implement and evaluate a digital solution to improve timely access to specialist advice', *International Journal of Integrated Care (IJIC)*, Vol. 5, No. 16, p.743.

- Nikkil, S. and Bijlsma, M.J. (2021) 'Educational note: causal decomposition of population health differences using Monte Carlo integration and the g-formula', *International Journal of Epidemiology*, Vol. 17, No. 6, p.6.
- Thompson, N., Odell-Miller, H., Underwood, B.R., Wolverson, E. and Hsu, M.H. (2024) 'How and why music therapy reduces distress and improves well-being in advanced dementia care: a realist review', *Nature Mental Health*, Vol. 2, No. 12, p.552.
- Thomsen, S.N., Lucia, A., Spence, R.R., Benatti, F.B., Joyner, M.J., Berg, R.M.G., Ried-Larsen, M. and Simonsen, C. (2024) 'First, do no harm: a call to action to improve the evaluation of harms in clinical exercise research', *British Journal of Sports Medicine*, Vol. 58, No. 12, p.3.
- Watson and Matthew (2018) 'Brexit, the left behind and the let down: the political abstraction of 'the economy' and the UK's EU referendum', *British Politics*, Vol. 20, No. 14, p.439.
- Zhao, H., Zheng, S., Wu, L., Yu, B. and Wang, J. (2024) 'LANE: logic alignment of non-tuning large language models and online recommendation systems for explainable reason generation', *Briefings in Functional Genomics*, Vol. 6, No. 14, p.837.