

Article

A Multi-Level Reinforcement Learning Reasoning Model Based on Adaptive Hierarchical Feature Fusion

Siyu Zhu ^{1,†} , Cheng He ^{2,†}, Yanlan Wang ¹, Qitao Tai ¹, Lin Miao ^{1,*}, Liang Wang ³, Xiulei Liu ¹, Yonghao Yang ¹, Jiahao Zhang ¹ and Weijian Guo ¹

¹ College of Computer Science, Beijing Information Science & Technology University (BISTU), Beijing 102200, China; zhushiyu@bistu.edu.cn (S.Z.); 2024020770@bistu.edu.cn (Y.Y.); 2025020708@bistu.edu.cn (J.Z.); 2025020769@bistu.edu.cn (W.G.)

² National Key Laboratory of Complex Aeronautical System Simulation, Beijing 100076, China; hechengtj@163.com

³ Shaanxi Aerospace Technology Application Research Institute Co., Ltd., Xi'an 710100, China; wangliang_fighting@163.com

* Correspondence: linmiao@bistu.edu.cn

† These authors contributed equally to this work.

Abstract

Current knowledge graph reasoning methods actually use the same state representations and a fixed decision method to complete the total reasoning process. They are not able to truly use the different levels of attribute information. To overcome this problem, we propose a multi-level reinforcement learning reasoning model based on adaptive hierarchical feature fusion. Our model divides attribute information into coarse-grained attributes and fine-grained attributes. In this way, our model can better balance overall semantics and details, thereby holding connections between entities and relationships. We design an adaptive hierarchical feature fusion mechanism. Our model dynamically determines whether it pays more attention to attribute information or relationship according to the needs of the current task, rather than constantly using the same weights. This enables our model to have better performance in different reasoning situations. We improve the decision process of the model, making the model know which relationships to focus on and which are less relevant to skip, thereby reducing ineffective search. According to the results of the experiment, our method is superior to the existing reinforcement learning reasoning methods, including MINERVA, AdaPath, CURL, and HMR. Our model improves MRR, Hits@1, Hits@3, and Hits@10 indicators compared with these baseline models. This shows that our method can improve the accuracy of inference and verify the effectiveness of our model in complex knowledge graph inference tasks.

Keywords: knowledge graph; reinforcement learning; hierarchical feature fusion; knowledge reasoning



Academic Editors: Zhixun Su, Yunyi Yan, Junxuan Wang and Lu Zhao

Received: 26 April 2026

Revised: 28 May 2026

Accepted: 25 June 2026

Published: 28 June 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Knowledge graphs organize entities, relations, and attributes into structured semantic networks and have become an important foundation for intelligent applications such as question answering, recommendation systems, industrial knowledge management, and decision support. With the increasing complexity of semantic association and relational inference tasks, knowledge graph reasoning has attracted significant attention in scenarios involving industrial maintenance, intelligent manufacturing, semantic retrieval, and expert

systems. The primary objective of knowledge graph reasoning is to infer latent relational knowledge from existing facts, thereby improving semantic completeness and reasoning capability. Although existing approaches have evolved from rule-based paradigms to representation learning frameworks, current methods still encounter difficulties in complex multi-hop reasoning scenarios, particularly in modeling hierarchical attribute semantics and maintaining reasoning efficiency.

In recent years, reinforcement learning has developed from single-agent models to multi-agent systems, providing new ideas for multi-hop reasoning in knowledge graphs. Multi-hop reasoning is a continuous process of making decisions that needs to consider both local information and global objectives. Based on this, we use hierarchical attribute modeling, adaptive feature fusion, and target-guided decision-making to enable our model to grasp the overall direction and pay attention to details. Hierarchical fusion and reinforcement learning perform well in complex decision tasks. Hierarchical fusion combines attribute features from different levels, enabling our model to better understand the semantics. The reinforcement learning constantly adjusts its decision method based environmental feedback. Dynamic event-triggered control and adaptive dynamic programming make decisions in continuous state and focus on whether the system can run stably. This is the difference between these methods and knowledge graph reasoning. Knowledge graph reasoning based on Reinforcement learning mainly studies how to make reasoning more efficient, semantic understanding more accurate, and strategy learning more stable. Existing reinforcement learning-based reasoning frameworks commonly formulate knowledge graph reasoning as a sequential path exploration problem. Representative methods, including DeepPath, MINERVA, AdaPath, CURL, and HMR, optimize reasoning policies through iterative interactions between reasoning agents and the knowledge graph environment. These approaches mainly emphasize dynamic path exploration and strategy learning based on current semantic states and historical reasoning paths, demonstrating strong capability in interpretable multi-hop reasoning. In addition to reinforcement learning, graph neural networks (GNNs) and Transformers have also become important methods in the field of knowledge graph reasoning. GNN is better at utilizing graph structure information to discover potential connections. The Transformer relies on self-attention mechanisms and pays more attention to global contextual information. The Transformer is able to capture long-distance dependency. These methods promote the development of knowledge graph reasoning.

Currently, there are several common methods for knowledge graph reasoning, including reinforcement learning, graph neural networks, and Transformers. These methods have their own advantages. However, they still have limitations. For example, reinforcement learning is suitable for making interpretable multiple inferences and dynamic decisions, but reinforcement learning cannot make enough use of hierarchical information in entity attributes. Hierarchical reasoning mainly divides paths into different levels or does the task step by step, but hierarchical reasoning does not handle the relationship between global information and fine-grained attributes well. Moreover, when selecting candidate actions, only the current state and historical information are usually considered, which can lead to repeated explorations in the large-scale search space and make strategy learning unstable.

To address these challenges, we propose a multi-level RL reasoning framework with adaptive hierarchical feature fusion. During multi-hop reasoning, our framework simultaneously captures coarse-grained global guidance and fine-grained discriminative semantic features via hierarchical attribute modeling, dynamically adjusts semantic interactions through an adaptive feature fusion mechanism, and constrains redundant action exploration using a target-guided action strategy. This integration enhances reasoning accuracy, efficiency, and stability, and is suitable for complex heterogeneous knowledge graph sce-

narios requiring hierarchical semantic modeling and multi-hop decision optimization, including industrial fault analysis, intelligent knowledge management, semantic retrieval, recommendation systems, and question answering.

The main contributions of our work are as follows:

1. A multi-level reinforcement learning reasoning framework is developed to incorporate hierarchical attribute information into a unified reasoning process, enhancing semantic representation capability and structural constraint modeling in knowledge graph reasoning.
2. An adaptive hierarchical feature fusion mechanism is introduced to dynamically coordinate hierarchical attribute features and relational semantics, improving semantic interaction modeling and reasoning adaptability across different tasks.
3. A target-guided action decision strategy is designed to constrain redundant action exploration and optimize multi-hop reasoning path selection through semantic-aware strategy learning.
4. Extensive comparative experiments and ablation studies are conducted to evaluate the effectiveness and superiority of our framework in knowledge graph reasoning and completion tasks.

2. Related Work

To comprehensively understand the current research landscape of knowledge graph reasoning, our work conducts a systematic survey from three perspectives: knowledge graph reasoning and representation learning methods, multi-hop reasoning and link prediction over knowledge graphs, and domain-specific knowledge graph reasoning and feature modeling, as detailed below:

2.1. Knowledge Graph Reasoning and Representation Learning Methods

Knowledge graphs organize entities, relations, and attribute information in a structured form and are widely applied in tasks such as question answering and recommendation systems. The goal of knowledge graph reasoning and completion is to infer potential relations based on existing triples, thereby enhancing the completeness and reasoning capability of the graph. Early studies primarily relied on rule-based reasoning methods, such as first-order predicate logic and ontology-based reasoning [1]. These approaches generate new knowledge through manually defined logical rules, exhibiting strong interpretability and consistency [2,3]. However, the construction of such rules depends heavily on human expertise, making them difficult to adapt to complex and dynamic relations, while also suffering from limited scalability in large-scale knowledge graphs [4,5]. For example, although the Ontological Pathfinding method improves efficiency through optimized rule mining, it still faces high computational complexity when applied to large-scale knowledge graphs [6]. With the advancement of representation learning methods, embedding-based knowledge graph reasoning has gradually become the mainstream approach. The TransE model [7] models entity relations through a translation mechanism, but it suffers from limited expressive capacity when handling complex relational patterns. The RotatE model [8] enhances the modeling of diverse relation patterns by performing rotational operations in the complex space, improving reasoning performance while preserving geometric interpretability. Furthermore, graph neural networks have been employed to capture neighborhood information in graph structures [9], and pre-trained language models have been introduced to improve completion performance [10]. Recent research has applied graph neural networks to multi-dimensional collaborative optimization.

Overall, rule-based reasoning methods exhibit advantages in structural constraints, while representation learning methods demonstrate superior scalability and efficiency.

Nevertheless, both paradigms face challenges in effectively integrating multi-hop path information and attribute features, thereby providing a foundation for the subsequent development of reinforcement learning-based approaches [11].

2.2. Multi-Hop Reasoning and Link Prediction over Knowledge Graphs

To address the problem of multi-hop reasoning, reinforcement learning (RL) has been introduced into knowledge graph reasoning tasks by modeling the reasoning process as a sequential path-search problem. Within a finite number of exploration steps, the reasoning strategy is optimized through environmental feedback, enabling dynamic modeling of complex relational patterns [12,13]. Representative RL-based reasoning methods exhibit distinct optimization focuses and reasoning paradigms. Early RL approaches primarily focused on the construction of path-search mechanisms. The DeepPath model [14] formulates multi-hop reasoning as a Partially Observable Markov Decision Process (POMDP) [15] and performs path expansion based on TransE embeddings, but it relies heavily on pre-training paths and incurs substantial computational overhead. The MINERVA model [16] incorporates Long Short-Term Memory (LSTM) networks [17] to encode historical paths, enabling goal-directed path search; however, its robustness remains limited under complex relational structures. The M-Walk model [18] combines Recurrent Neural Networks (RNNs) with Monte Carlo Tree Search (MCTS) [19] to alleviate the sparse reward problem, yet its computational complexity scales quadratically with the growth of graph scale. The development of MAS-RL [20] has inspired knowledge graph reasoning. The ideas, including value decomposition, consensus learning, and adaptive hierarchical policies, can help multiple inference agents better complete multi-hop path exploration. The Sec-GNN [21] uses a graph neural network to achieve multi-task optimization in drone satellite communication. The dynamic authority allocation-based event-triggered control framework [22] improves the efficiency of cooperation between humans and machines by dynamically adjusting authorization and triggering mechanisms. The adaptive dynamic programming-based event-triggered consensus approach [23] combines distributed learning and enables different intelligent agents to work together. The advanced RL methods mentioned above have demonstrated substantial feasibility for multi-hop reasoning; nevertheless, a series of advancements have emerged from the perspectives of strategy optimization and structural enhancement. AdaPath [24] introduces adaptive path memory mechanisms to enhance long-range dependency retention and dynamic path evolution; however, its optimization objective mainly concentrates on memory adaptation instead of semantic hierarchy construction. The CURL model [25] reduces reasoning redundancy through collaborative dual-agent optimization and action space exploration strategies, but it still relies on flattened semantic representations during strategy learning. The HMR model [26] introduces hierarchical reasoning structures to decompose reasoning stages and improve reasoning efficiency; nevertheless, its hierarchy is primarily reflected in the reasoning process itself rather than the hierarchical semantic organization of entity attributes. Further studies have explored alternative directions to improve reasoning performance, such as context-aware meta-reinforcement learning (CAMRL) [27], soft-reward optimization (Multi-Hop) [28], and multi-modal integration (MMKGR) [29], alongside other representative frameworks including KGQA [30], RLH [31], and SBS [32]. Apart from reinforcement learning, Graph Neural Networks (GNNs) and Transformers are widely used for knowledge graph reasoning and link prediction. The CompGCN [33] introduces a composition operation in graph convolutional networks to jointly embed entities and relationships. The NBFNet [34] extends the classical Bellman-Ford algorithm into a path-based GNN framework for extensible link prediction. To handle inductive reasoning tasks, GraIL [35] extracts local subgraphs to predict relations without relying on entity identity information. Concur-

rently, Transformer-based methodologies, represented by KG-BERT [36], use pre-trained language models to encode entities and relations into text descriptions, thereby learning deep semantic representations based on context.

Although these methods improve knowledge graph reasoning capabilities from different perspectives, critical limitations still remain within current frameworks. Existing reinforcement learning reasoning methodologies generally focus on relation path traversal while insufficiently using the semantic constraints carried by entity attributes. Attribute information is routinely treated as flat semantic features without distinguishing hierarchical semantic dependencies, which weakens semantic discrimination and feature alignment capabilities during multi-hop reasoning. In addition, current reasoning frameworks usually lack explicit target-oriented semantic constraints during action selection, resulting in redundant path exploration, severe strategy fluctuation, and unstable reasoning behaviors in large-scale action spaces. The Sec-GNN and dynamic control mechanisms also use reinforcement learning and optimize with a graph, but there is a fundamental difference between these methods and knowledge graph reasoning. In the complex multi-hop reasoning tasks, GNNs and Transformers present several problems. Firstly, they usually regard link prediction as a neighborhood aggregation or a black-box classification problem, so they can not explain the reasoning path. Secondly, they lack effective control over the dynamic action space. Thirdly, in dense graph structures or large-scale relational search spaces, they are prone to high computational costs, and their ability to represent is easily diluted. Furthermore, reinforcement learning used for knowledge graph multi-hop inference has a theoretical convergence basis. Limited search space and target constraints ensure the stability of the learning process. The reward mechanism leads the agent to find a better path. Hierarchical feature fusion reduces surplus information and accelerates model convergence. These mechanisms provide stability guarantees for multi-hop path optimization.

These methods mainly make decisions in a continuously changing environment, considering how to ensure communication security, maintain system stability, enhance collaborative effects, and reduce control costs. Our work studies multi-hop inference on discrete relationship graphs. We use perceived attention and hierarchical semantic constraints to reduce repetitive exploration, improving decision consistency and boosting reasoning accuracy and reasoning efficiency.

2.3. Domain-Specific Knowledge Graph Reasoning and Feature Modeling

In practical application scenarios, knowledge graphs typically contain rich attribute information and complex relational structures. Existing studies have made notable progress in knowledge graph construction and application. For instance, multimodal knowledge graph approaches enhance knowledge representation by integrating multi-source data [37]. In addition, several works have validated the effectiveness of knowledge graphs in information systems and data management contexts [38,39]. At the reasoning level, some studies still rely primarily on rule-based methods and have achieved knowledge reasoning in specific scenarios [40,41]. Meanwhile, approaches incorporating large language models further improve reasoning capabilities in complex tasks [42]. Furthermore, knowledge graphs have demonstrated potential value in system analysis and causal relationship discovery [43].

However, a comprehensive review of existing studies reveals several limitations. First, attribute information is typically modeled in a coarse-grained manner, lacking explicit characterization of hierarchical structures and attribute heterogeneity, which makes it difficult to provide effective constraints for the reasoning process. Second, relations are often treated equally during reasoning without distinguishing their different strengths of association, resulting in a redundant action space. Third, existing reasoning strategies lack task-specific adaptability, making it challenging to balance reasoning accuracy and efficiency.

3. Methods

Based on the analysis of existing knowledge graph reasoning methods, several limitations can be identified in hierarchical attribute modeling, reasoning efficiency, and target-aware path optimization. To address these issues, a multi-level reinforcement learning reasoning framework based on adaptive hierarchical feature fusion is developed. Figure 1 presents the overall architecture of our framework for knowledge graph reasoning. Our framework integrates hierarchical semantic representation learning with reinforcement learning-based strategy optimization and consists of four main components: the multi-dimensional state representation layer, the adaptive hierarchical feature fusion layer, the action decision layer, and a reward mechanism for reasoning optimization.

At the input stage, the knowledge graph is modeled as the reasoning environment, where the current entity, candidate relations, and historical reasoning paths are encoded into unified state representations. The multi-dimensional state representation layer projects entities and relations into a continuous semantic embedding space and incorporates sequential path modeling to capture contextual dependencies during multi-hop reasoning. As a result, the generated state representations preserve structural semantic information and historical reasoning context for subsequent strategy learning and action decision-making.

The adaptive hierarchical feature fusion layer is designed to model semantic interactions between hierarchical attribute features and relational representations. Specifically, macro-level and micro-level attribute semantics are encoded through a hierarchical representation mechanism and dynamically integrated through adaptive gating operations. During reasoning, the fusion process adjusts semantic contributions according to the current reasoning state and path context. Macro-level semantic features provide global guidance in the early reasoning stages, whereas micro-level attribute features enhance fine-grained semantic discrimination in later stages. This hierarchical fusion strategy enables progressive semantic refinement throughout the multi-hop reasoning process.

In the action decision layer, a query-aware attention mechanism is introduced to evaluate the semantic relevance between candidate actions and reasoning targets. The attention-enhanced semantic representations are further combined with the reinforcement learning policy network to guide action selection during multi-hop reasoning. By assigning higher policy probabilities to semantically relevant actions, our framework reduces redundant exploration within large candidate action spaces while preserving semantic consistency along reasoning paths.

For reasoning optimization, a multi-dimensional reward mechanism is constructed to jointly evaluate reasoning accuracy, path efficiency, and semantic evolution consistency. The reward signals provide policy feedback during reinforcement learning training, encouraging the reasoning agent to generate semantically reasoning paths while avoiding redundant path exploration.

Our framework differs significantly from dynamic event-triggered control for human-machine cooperative systems and the adaptive dynamic programming-based event-triggered consensus approach. These methods mainly face continuous control scenes, where state change is decided by system dynamics, and action can be understood as how to control the system or when to trigger a strategy update. These methods mainly solve the system stability, consistency, and communication cost reduction problem. Our framework is designed for knowledge graph multi-hop reasoning, where the reasoning environment is composed of discrete relationship graphs. Action decides which relationship path to take next and how to make a reasoning decision. In the state representation, the dynamic control method mainly uses continuous state and system observation information. Our work considers entity, relation, hierarchical attribute, and existing reasoning process information together. The hierarchical feature fusion combines the overall semantics with

local attribute information. The adaptive gating mechanism adjusts the importance of different features based on the current reasoning goal. In the decision mechanisms and optimization objectives, the adaptive dynamic programming method studies continuous control strategies. The query-aware attention mechanism in our work chooses a more relevant relationship with the goal, guiding path search. Thus, our work is not studying how to make a control system more stable, but emphasizes relation path search, association modeling, and hierarchical attribute constraint.

To get better reasoning strategies, our work designs the hierarchical feature fusion, adaptive gating, query-aware attention, and multi-dimensional reward mechanisms. The purpose is to enable the model to better understand semantic information and make more accurate decisions in the reasoning process. Our work combines multiple semantic interaction modules, making the entire learning and optimization process more complex, making it is difficult to strictly ensure convergence in theory. To achieve better convergence, we observed performance in the training process and conducted different rounds of experiments under different random seeds. Comparative experiments were conducted in different reasoning environments to verify the effectiveness and stability of the model. Experimental results show that our framework is stable in the training process, and has consistent reasoning effects. Further convergence analysis in theory will be conducted in subsequent work.

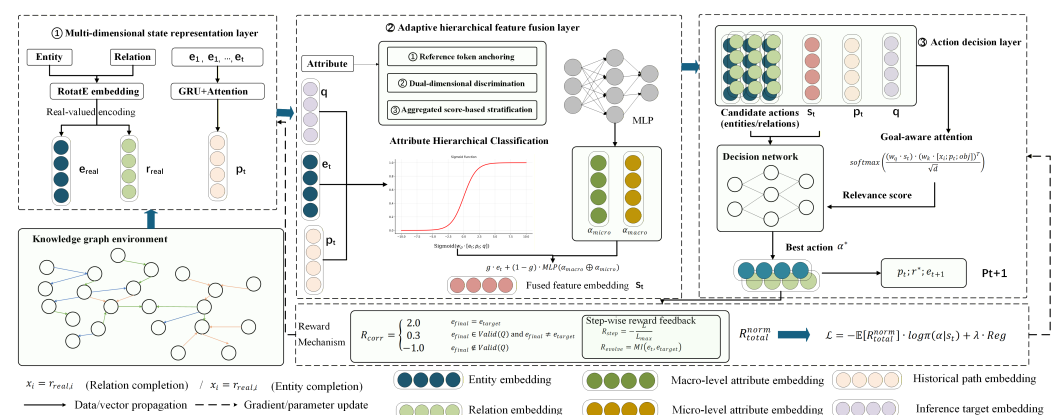


Figure 1. Overall framework of our multi-level reinforcement learning reasoning model based on adaptive hierarchical feature fusion. Our framework consists of three main components: the multi-dimensional state representation layer, which encodes entities, relations, and historical reasoning paths into unified semantic representations; the adaptive hierarchical feature fusion layer, which dynamically integrates hierarchical attribute and relation features for semantic interaction modeling; and the action decision layer, which performs query-guided action evaluation and reinforcement learning-based reasoning path optimization.

Overall, our framework achieves unified modeling of structural semantics, hierarchical attribute interactions, and reinforcement learning-based policy optimization in knowledge graph reasoning. Through the coordinated integration of state representation, hierarchical feature fusion, action decision, and reward feedback mechanisms, our framework improves both reasoning accuracy and reasoning efficiency in complex multi-hop reasoning scenarios. To improve the readability of the mathematical formulation, the primary symbols and variables used in our framework are summarized in Table 1. The notation table provides unified definitions for state representations, hierarchical feature fusion operations, query-aware action decision mechanisms, and reward optimization functions employed in our framework.

Table 1. Notation and symbol definitions used in our reasoning model.

Symbol	Description
x_i	Concatenated semantic representation of the i -th reasoning path node, consisting of entity embedding e_i and relation embedding r_i
h_i	Hidden semantic state generated by the GRU encoder at reasoning step i for sequential path dependency modeling
α_i	Normalized attention coefficient of the i -th hidden state computed through the Softmax attention mechanism
w_h	Trainable semantic attention parameter vector used for hidden state importance estimation
p_t	Aggregated path semantic representation generated through weighted attention summation over sequential hidden states
$\bar{n}$	Global macro-semantic anchor representation obtained from cross-entity semantic prototype aggregation
$sim_{sem}(i)$	Semantic cosine similarity between attribute representation n_i and the macro-semantic anchor center $\bar{n}$
H_i	Information entropy measuring semantic distribution uncertainty of attribute i
$cov(i)$	Coverage ratio of attribute i over all entities within semantic category k
$score(i)$	Hierarchical semantic discrimination score utilized for macro-level and micro-level attribute partitioning
θ_k	Adaptive semantic partition threshold for distinguishing macro-level and micro-level attributes within semantic category k
a_{macro}	Encoded macro-level semantic attribute representation used for global reasoning guidance
a_{micro}	Encoded micro-level semantic attribute representation used for fine-grained semantic discrimination
g	Adaptive semantic gating coefficient controlling the fusion contribution of entity and attribute representations
s_t	Unified reinforcement learning state representation integrating entity semantics, path context, and hierarchical attribute information
$Score_i$	Query-guided semantic relevance score assigned to the i -th candidate action during action evaluation
$\pi(a s_t)$	Reinforcement learning policy distribution over candidate reasoning actions conditioned on state s_t
$Q_{net}(s_t, a)$	Estimated action-value function produced by the Deep Q-Network for state-action pair (s_t, a)
h_q	Intermediate hidden semantic representation generated by the first fully connected layer in the DQN module
$\pi'(a s_t)$	Adjusted policy distribution after semantic constraint refinement and action perturbation
a^*	Final optimal reasoning action selected through maximum policy probability evaluation
R_{total}	Overall reinforcement learning reward integrating reasoning correctness, path efficiency, and semantic evolution consistency
R_{corr}	Reward component evaluating semantic correctness between predicted entities and target entities
R_{step}	Reasoning step penalty term designed to suppress redundant path traversal and inefficient exploration
L	Total number of sequential reasoning actions within the current multi-hop reasoning trajectory
R_{evolve}	Semantic evolution reward measuring progressive consistency toward target-oriented reasoning objectives
R_{total}^{norm}	Normalized global reward signal constrained within $[0, 1]$ to improve reinforcement learning stability
$\mathcal{L}$	Optimization objective function utilized for reinforcement learning policy update and gradient backpropagation

3.1. Multi-Dimensional State Representation Layer

The multi-dimensional state representation layer first employs the RotatE model to initialize entity and relation embeddings. Specifically, the entity set E and relation set R

are projected into the complex vector space $\mathbb{C}^d$. For any head entity $h \in E$, tail entity $t \in E$, and relation $r \in R$, the relational transformation is formulated as:

$$t = h \circ r \quad (1)$$

where $\circ$ denotes the element-wise Hadamard product. Each dimension of the relation embedding r satisfies the unit modulus constraint $|r_i| = 1$, allowing relational transformations to be represented as phase rotations in the complex space. This formulation effectively captures structural properties such as symmetry and anti-symmetry in knowledge graph relations.

To model sequential dependencies in multi-hop reasoning paths, a Gated Recurrent Unit (GRU) combined with an attention mechanism is introduced to encode the historical reasoning path $P_t = [e_1, r_1, e_2, r_2, \dots, e_t]$. The entity-relation pairs in the reasoning path are connected to construct the sequential input representations:

$$x_i = [e_i; r_i], \quad i = 1, 2, \dots, t-1 \quad (2)$$

The GRU is employed to capture contextual dependencies along the reasoning path, and the hidden state update process is defined as:

$$h_i = \text{GRU}(x_i, h_{i-1}) \quad (3)$$

where h_0 represents the zero-initialized hidden state. To emphasize the historical states that are most relevant to the current reasoning decision, an attention mechanism is further incorporated. The attention weight α_i is calculated as:

$$\alpha_i = \text{Softmax}\left(\frac{w_h \cdot h_i}{\sqrt{d}}\right) \quad (4)$$

where $w_h \in \mathbb{R}^d$ denotes a trainable attention parameter vector. Based on the computed attention weights, the final reasoning path representation p_t is obtained through weighted aggregation of the hidden states:

$$p_t = \sum_{i=1}^{t-1} \alpha_i \cdot h_i \quad (5)$$

Finally, the multi-dimensional state representation layer outputs unified semantic representations consisting of the entity embedding e , relation embedding r , and reasoning path encoding p_t . These representations provide the semantic foundation for subsequent hierarchical feature fusion and policy learning.

3.2. Adaptive Hierarchical Feature Fusion Layer

The adaptive hierarchical feature fusion layer first performs automatic hierarchical partitioning of attribute triples in the knowledge graph. Five cross-entity semantic anchor categories, including category, usage, description, function type, and constraint level, are selected as macro-level semantic references. The selection of these anchors is not based on experience, but is theoretically motivated by hierarchical semantic organization and semantic information representation requirements in multi-hop knowledge graph reasoning. The coarse-grained anchors capture global semantic constraints, keeping semantic consistency in long-distance reasoning. The fine-grained anchors focus on local attribute difference, helping the model to distinguish tiny differences between the candidate entity and relation. In the whole, the hierarchy design combines global semantic and local information, improving semantic discrimination ability and feature fusion effect. The anchor-guided partitioning method reduces semantic confusion caused by attributes of different structures. The overall design considers the semantic level, attribute association, and contribution of

multi-hop reasoning, providing guides and evidence for the subsequent refinement of the division globally.

A pre-trained BERT encoder is employed to generate semantic embeddings $n_1, \dots, n_5$ for the anchor categories. The average semantic anchor vector $\bar{n}$ is defined as:

$$\bar{n} = \frac{1}{5} \sum_{j=1}^5 n_j \quad (6)$$

It is important to note the methodological distinction between this adaptive hierarchical fusion mechanism and control paradigms commonly employed in other domains, such as dynamic event-triggered control and adaptive dynamic programming. Control methods usually reduce unnecessary control updates by a threshold, with the goal of system stability or achieving the best overall performance in a continuous dynamic environment. Their goal is not semantic reasoning. Our work is used for knowledge graph reasoning, dealing with discrete and multi-level entities and attributes, directly affecting how to choose a multi-hop path. The fusion strategy combines global semantic and local information, improving stability of reasoning and distinguishability. Our work is used for semantic understanding and path selection, instead of traditional system control optimization.

Based on semantic anchor representations, hierarchical attribute partition is executed by integrating semantic similarity analysis with attribute value distribution modeling. Attributes having high semantic similarity to anchor vectors and strong cross-entity coverage are allocated to coarse-grained macro-level sets to enforce global semantic constraints. Conversely, attributes characterized by lower similarity or more specialized distributions are assigned to fine-grained micro-level sets, thereby supplying cues for precise entity matching and relation filtering. This coarse-to-fine partitioning strategy ensures that hierarchical semantics are explicitly structured to promote adaptive feature fusion during inference. The semantic similarity between the attribute name embedding n_i and the anchor center vector is computed using cosine similarity:

$$\text{sim}_{\text{sem}}(i) = \frac{n_i \cdot \bar{n}}{\|n_i\| \cdot \|\bar{n}\|} \quad (7)$$

In addition, attribute entropy H_i and coverage ratio $\text{cov}(i)$ are introduced to characterize the statistical distribution of attribute values:

$$H_i = - \sum_{v \in V_i} p_v \log p_v \quad (8)$$

$$\text{cov}(i) = \frac{N_{\text{has},i}}{N_{\text{total},k}} \quad (9)$$

where V_i denotes the value set of attribute i , and $N_{\text{total},k}$ represents the total number of entities within the k -th semantic category. Based on the above semantic and statistical metrics, a comprehensive hierarchical discrimination score $\text{score}(i)$ is formulated as:

$$\text{score}(i) = 0.4 \times \text{sim}_{\text{sem}}(i) + 0.3 \times (1 - H_{\text{norm}}(i)) + 0.3 \times \text{cov}_{\text{norm}}(i) \quad (10)$$

A category-specific threshold θ_k is introduced for hierarchical attribute partitioning. If $\text{score}(i) \geq \theta_k$, the attribute is assigned to the macro-level attribute set $\mathcal{A}_{\text{macro}}$; otherwise, it is divided into the micro-level attribute set $\mathcal{A}_{\text{micro}}$.

In multi-hop reasoning scenarios, hierarchical attribute semantics contribute differently to semantic decision-making. Macro-level attributes mainly provide global semantic guidance and category-level semantic constraints, whereas micro-level attributes supply

discriminative semantic cues for fine-grained entity matching and relation filtering. Conventional flat feature modeling strategies usually treat heterogeneous attributes uniformly, which weakens semantic discrimination and may introduce noisy reasoning behaviors during reinforcement learning-based reasoning.

To obtain unified semantic representations for hierarchical attributes, a Multi-Layer Perceptron (MLP) is employed to encode macro-level and micro-level attribute features into d -dimensional embedding vectors:

$$a_{macro} = \sigma(W_m \cdot [a_n; a_v] + b_m) \quad (11)$$

$$a_{micro} = \sigma(W_s \cdot [a_n; a_v] + b_s) \quad (12)$$

where a_n and a_v denote the initial embeddings of the attribute name and attribute value, respectively, $[\cdot; \cdot]$ represents the vector concatenation operation, and $W_m, W_s \in \mathbb{R}^{d \times 2d_a}$ are learnable transformation matrices.

To dynamically coordinate hierarchical semantic information during multi-hop reasoning, an adaptive gating unit g is introduced for semantic-aware feature fusion. It is different from fixed weight. The gating unit adjusts the importance of different hierarchical features based on the current reasoning state and the semantic context. A semantic anchor is like a reference object. Macro attributes closer to the anchor provide overall direction in the early reasoning stage. Micro attributes that have differences in details adjust weight in the subsequent reasoning stage to help the model distinguish between the candidate entity and the relationship. All attributes are not equally important, and it is necessary to dynamically balance macro attributes and micro attributes. In this way, our model can keep semantic consistency and make more accurate and more robust judgments in a complex reasoning scene. Different from conventional static gating or attention mechanisms, our adaptive fusion strategy adjusts semantic contributions according to the current reasoning state, historical reasoning trajectory, and target semantics. From a theoretical perspective, the hierarchical attribute division informed by anchors improves semantic separability, reduces noisy exploration in reinforcement learning, and ensures that multi-hop reasoning decisions are guided by both coarse-grained global context and fine-grained discriminative features. This validates the rationality and necessity of introducing hierarchical anchors in multi-level feature fusion. The fused state representation s_t is formulated as:

$$s_t = g \cdot e_t + (1 - g) \cdot \text{MLP}(a_{macro} \oplus a_{micro}) \quad (13)$$

where e_t denotes the current entity embedding and $\oplus$ represents vector concatenation. The adaptive gating coefficient $g \in [0, 1]$ is jointly determined by the current reasoning state and the semantic reasoning objective:

$$g = \text{Sigmoid}(w_g \cdot [e_t; p_t; q]) \quad (14)$$

where p_t denotes the historical reasoning path representation and q represents the semantic embedding of the reasoning target. The sigmoid activation function is employed to normalize the gating coefficient into the interval $[0, 1]$, enabling adaptive semantic weighting and feature filtering during hierarchical feature fusion. Moreover, the smooth nonlinear characteristics of the sigmoid function contribute to stable gradient propagation during reinforcement learning optimization and relieve training instability caused by noisy semantic interactions. The gating parameter w_g is initialized using random uniform initialization, and the hidden representation dimension is aligned with the semantic embedding dimension d to preserve semantic consistency during feature interaction and strategy optimization.

Through adaptive coordination between macro-level semantic guidance and micro-level discriminative semantic features, our mechanism enhances semantic separability in reinforcement learning states and reduces ineffective action exploration during multi-hop reasoning. The adaptive hierarchical feature fusion layer outputs the fused semantic representation $s_t \in \mathbb{R}^d$, which serves as the input representation for the subsequent action decision layer.

3.3. Action Decision Layer

The action decision layer introduces a query-aware attention mechanism, which is used to check how closely the candidate action is related to the reasoning target. When performing multi-hop reasoning in reinforcement learning, it is necessary to choose from a lot of candidate relations. Traditional methods mainly rely on the current state and historical path, and it is easy to get an erroneous path or repeatedly search. Thus, our work regards the query target as dynamic guidance, making the model make a decision towards the final target, reducing invalid search, and avoiding the wrong direction of reasoning. The fused state representation s_t captures the semantic context of the current reasoning stage, whereas the candidate action embedding x_i encodes the semantic characteristics of potential reasoning transitions. In addition, the reasoning objective vector obj provides target-oriented semantic guidance. The query-aware attention mechanism combines current semantic information, dynamically evaluating the related degree between the candidate action and the target, maintaining more related actions, and reducing invalid transfer and repetitive search.

Based on the semantic interactions between the current reasoning state and candidate actions, our mechanism performs soft filtering over the candidate action space. The semantic relevance score $Score_i$ is calculated as:

$$Score_i = \text{Softmax} \left(\frac{(W_q \cdot s_t) \cdot (W_k \cdot [x_i; p_t; obj])^\top}{\sqrt{d}} \right) \quad (15)$$

where $W_q, W_k \in \mathbb{R}^{d \times d}$ denote trainable attention projection matrices, s_t represents the fused semantic state representation, x_i denotes the candidate action embedding, p_t is the historical reasoning path representation, and obj represents the semantic embedding of the reasoning objective. All semantic representations are projected into the same hidden semantic dimension d . The scaling factor $\sqrt{d}$ is introduced to relieve excessively concentrated attention distributions caused by large inner-product values in high-dimensional semantic spaces, thereby improving numerical stability during attention computation and reinforcement learning optimization.

The hidden semantic dimension d is consistently shared across the query representation, historical reasoning path, and candidate action embeddings to preserve semantic consistency during feature interaction. The trainable projection matrices W_q and W_k map heterogeneous semantic representations into a unified latent attention space, improving semantic alignment between reasoning states and candidate actions. Dot-product similarity is adopted due to its computational efficiency and effectiveness in measuring semantic correlations in high-dimensional embedding spaces. Combined with Softmax normalization, the query-aware attention mechanism dynamically emphasizes semantically relevant actions while suppressing redundant or low-relevance candidates, thereby improving semantic focus and reasoning stability during multi-hop reasoning.

The semantic relevance scores are further incorporated into the strategy network to guide the reasoning agent toward actions with high target relevance. The policy function $\pi(a|s_t)$ is defined as:

$$\pi(a|s_t) = \text{Softmax}(Q_{net}(s_t, a) + \gamma \cdot \text{Score}_i) \quad (16)$$

where $\gamma \in [0, 1]$ denotes the weighting coefficient of the semantic relevance score. The action-value function $Q_{net}(s_t, a)$ is estimated using a Deep Q-Network (DQN) with a two-layer fully connected architecture:

$$h_q = \text{ReLU}(W_1 \cdot [s_t; a] + b_1) \quad (17)$$

$$Q_{net}(s_t, a) = W_2 \cdot h_q + b_2 \quad (18)$$

where $W_1 \in \mathbb{R}^{2d \times 2d}$, $b_1 \in \mathbb{R}^{2d}$, $W_2 \in \mathbb{R}^{1 \times 2d}$, and $b_2 \in \mathbb{R}$ are trainable network parameters. For optimal action selection, the reasoning agent identifies the action a^* according to the learned strategy distribution. To avoid deterministic action selection when multiple candidate actions provide similar policy scores, a Gaussian perturbation $\epsilon \sim \mathcal{N}(0, \sigma^2)$ is introduced:

$$\pi'(a|s_t) = \pi(a|s_t) + \epsilon \quad (19)$$

The final action decision rule is formulated as:

$$a^* = \arg \max_a \pi'(a|s_t) \quad (20)$$

The reasoning process terminates when the policy probability satisfies $\pi(a_i|s_t) \geq \tau$ or the maximum reasoning step limit L_{max} is reached. Finally, the action decision layer outputs the optimal action and the updated reasoning path $P_{t+1} = [P_t; r^*; e_{t+1}]$.

3.4. Reward Mechanism

To make the model more accurate and train more stably, our work designs a multi-dimensional reward mechanism, which considers result correctness, inference efficiency, semantic consistency, and path rationality at the same time. In this way, our mechanism reduces common problems in multi-hop reasoning in reinforcement learning. For example, different rewards are responsible for different tasks. The accuracy reward checks if the final result is right, and the step penalty avoids the model taking too many invalid paths. The semantic evolution reward guides reasoning towards the target. Our model can learn more stable and correct decision strategies by combining these rewards.

Accordingly, the total reward R_{total} is defined as a weighted combination of the accuracy reward R_{corr} , step penalty R_{step} , and semantic evolution reward R_{evolve} . From a theoretical perspective, the weighted scoring mechanism is designed to model the heterogeneous contribution of different semantic features to reasoning decisions. Since semantic attributes show varying degrees of relevance, discriminative ability, and structural importance during multi-hop reasoning, uniform treatment of all features may weaken semantic focus and introduce noisy optimization signals. Therefore, dynamic weight allocation is employed to strengthen high-relevance semantic information while suppressing low-contribution or redundant semantic patterns. This mechanism establishes an adaptive feature contribution framework that aligns semantic importance with decision influence, thereby improving semantic discrimination and reasoning consistency throughout the reinforcement learning process. Furthermore, the weighted allocation strategy enhances decision optimization by enabling the reasoning agent to prioritize semantically critical transitions and reduce ineffective exploration behaviors in large-scale action spaces.

$$R_{total} = \omega_1 \cdot R_{corr} + \omega_2 \cdot R_{step} + \omega_3 \cdot R_{evolve} \quad (21)$$

where the weighting coefficients ω_1 , ω_2 , and ω_3 are empirically set to 0.5, 0.2, and 0.3, respectively. These values are set according to the importance and dependency of rewards in multi-hop reasoning. The accuracy reward weight is the highest and is used to highlight the importance of the final result. The semantic evolution reward weight is moderate and is used to guide state keep consistency during the process. The step penalty weight is small and is used to limit path length without excessively affecting search. The weight distribution provides evidence for feature fusion, making different features dynamically adjust their contribution based on the effect. The accuracy reward weight ensures the main direction. The semantic evolution reward provides semantic guidance for the process and reduces the reward sparsity problem. The smaller step penalty weight avoids limiting search ability too much. Overall, the weight method establishes a mapping relationship between the importance of attributes and the contribution of features, making the model inclined to use key semantic features and reducing irrelevant information interference. Our model balances between accuracy, consistency, and efficiency, thereby improving reasoning stability and effectiveness.

The accuracy reward R_{corr} evaluates the matching degree between the final inferred entity e_{final} and the target entity e_{target} to guide our model toward effective reasoning trajectories. The reward formulation is defined as:

$$R_{corr} = \begin{cases} 2.0 & e_{final} = e_{target} \\ 0.3 & e_{final} \in Valid(Q) \text{ and } e_{final} \neq e_{target} \\ -1.0 & e_{final} \notin Valid(Q) \end{cases} \quad (22)$$

where $Valid(Q)$ denotes the set of contextually valid candidate entities corresponding to the reasoning query Q . The reward levels in Equation (22) provide different optimization signals for different reasoning outcomes. A large positive reward is assigned when the inferred entity exactly matches the target entity, encouraging successful reasoning paths. A smaller positive reward is provided for semantically valid but non-target entities to preserve exploration flexibility and relieve excessively sparse reward feedback. Conversely, invalid reasoning results receive negative penalties to suppress incorrect semantic transitions and noisy reasoning behaviors.

The hierarchical reward assignment enables our model to maintain both exploration capability and target-oriented optimization during reinforcement learning training. In addition, the negative penalty mechanism constrains unreasonable path expansion and relieves strategy instability caused by repeated invalid exploration. To suppress redundant reasoning detours and improve reasoning efficiency, a step penalty term is introduced:

$$R_{step} = -\frac{L}{L_{max}} \quad (23)$$

where L denotes the number of action steps in the current reasoning path, and L_{max} represents the predefined maximum allowable path length. The step penalty functions as a path-length regularization mechanism to suppress inefficient reasoning paths and reduce redundant multi-hop exploration. Without this constraint, reinforcement learning agents may fall into cyclic reasoning behaviors or excessively long reasoning chains, increasing computational overhead and error propagation risk in large-scale knowledge graphs.

The semantic evolution reward R_{evolve} employs Mutual Information (MI) to quantify the semantic correlation between the current entity $\mathbf{e}_t$ and the target entity $\mathbf{e}_{target}$, guiding the reasoning trajectory toward the target objective:

$$R_{evolve} = MI(\mathbf{e}_t, \mathbf{e}_{target}) \quad (24)$$

This function takes real-valued entity embeddings as input and outputs a reward value $R_{evolve} \in [0, 1]$. Compared with terminal-only reward supervision, the semantic evolution reward provides continuous optimization guidance during multi-hop reasoning and effectively alleviates sparse reward distribution problems in reinforcement learning. By measuring semantic consistency between intermediate reasoning states and the target entity, our model gradually optimizes the reasoning direction before reaching the final target entity.

Furthermore, the semantic evolution reward improves policy smoothness and reduces sudden policy fluctuations during reinforcement learning optimization. Since semantic similarity changes continuously along reasoning paths, the reward signal becomes more stable and informative than purely discrete terminal rewards.

To improve training stability, the total reward is normalized to obtain the final reward signal R_{total}^{norm} :

$$R_{total}^{norm} = \frac{R_{total} - R_{total}^{min}}{R_{total}^{max} - R_{total}^{min}} = \frac{R_{total} + 0.7}{2.0} \quad (25)$$

The reward normalization process aims to maintain numerical stability during reinforcement learning optimization and prevent unstable policy updates caused by unbalanced reward scales. Since different reward components have different numerical ranges and semantic meanings, direct accumulation may introduce optimization bias and training instability. Therefore, normalization is employed to balance the influence of multiple reward signals and maintain stable gradient propagation during policy learning.

Moreover, the reward balancing mechanism improves coordination among target correctness supervision, semantic guidance, and path efficiency optimization, reducing noisy exploration behaviors and alleviating strategy fluctuation problems in large-scale action spaces.

The model parameters are updated via policy gradient backpropagation. The loss function $\mathcal{L}$ is defined as:

$$\mathcal{L} = -\mathbb{E}_{\pi}[R_{total}^{norm} \cdot \log \pi(a|s_t)] + \lambda \cdot \|\theta\|^2 \quad (26)$$

where λ denotes the regularization coefficient, $\pi(a|s_t)$ represents the policy distribution for selecting action a under state s_t , and $\|\theta\|^2$ denotes the L_2 regularization term. The normalized reward signal is propagated through policy gradient optimization to guide adaptive optimization of the reasoning strategy.

Overall, the multi-dimensional reward mechanism establishes collaborative optimization among target supervision, semantic evolution guidance, path efficiency control, and penalty constraints. Through balanced reward allocation and normalized reinforcement learning optimization, our framework improves reasoning stability, suppresses blind exploration behaviors, and enhances convergence robustness in complex multi-hop knowledge graph reasoning scenarios. Experimental observations further demonstrate that the reward design improves reasoning consistency and policy stability under different random initialization conditions and reasoning environments.

3.5. Complexity Analysis

To evaluate the computational efficiency and scalability of our framework, we analyze the time and space complexities of the major components, including the adaptive hierarchical feature fusion module, the query-aware attention mechanism, the multi-hop reasoning process, and the action decision network.

Let d denote the hidden semantic dimension, $|\mathcal{A}|$ denote the average number of candidate actions at each reasoning step, L denote the reasoning path length, and N_a denote the number of entity attributes.

For the adaptive hierarchical feature fusion module, the primary computational cost arises from hierarchical semantic encoding and adaptive gating operations. The MLP-based attribute encoding process projects both macro-level and micro-level attribute representations into a unified semantic space, resulting in the following computational complexity:

$$O(N_a \cdot d^2) \quad (27)$$

The adaptive gating unit introduces an additional computational cost of:

$$O(d) \quad (28)$$

since the gating coefficient is generated through a lightweight sigmoid-based projection operation.

For the query-aware attention mechanism, the computational complexity mainly depends on semantic relevance estimation between the current reasoning state and candidate actions. Given the candidate action set $\mathcal{A}$, the attention computation complexity at each reasoning step is defined as:

$$O(|\mathcal{A}| \cdot d) \quad (29)$$

where the scaled dot-product attention operation evaluates semantic similarity between the query representation and candidate action embeddings.

During multi-hop reasoning, the overall reasoning complexity increases with the reasoning depth L . Without effective action constraints, exhaustive path exploration may result in exponential search complexity:

$$O(|\mathcal{A}|^L) \quad (30)$$

which substantially increases computational cost in large-scale knowledge graphs.

To alleviate this issue, our framework incorporates target-guided action constraints and query-aware action pruning mechanisms. By dynamically filtering semantically irrelevant candidate actions according to reasoning objectives, the effective action space is reduced from $|\mathcal{A}|$ to a smaller candidate subset $|\mathcal{A}'|$, where $|\mathcal{A}'| \ll |\mathcal{A}|$. Consequently, the reasoning complexity becomes:

$$O(|\mathcal{A}'|^L) \quad (31)$$

thereby significantly reducing redundant path exploration and improving reasoning efficiency.

In the action decision layer, the Deep Q-Network (DQN) mainly introduces computational cost associated with fully connected transformations. Since the strategy network adopts a dual-layer architecture with hidden dimension $2d$, the computational complexity of action evaluation for each candidate action is approximately:

$$O(d^2) \quad (32)$$

Regarding space complexity, the primary memory consumption originates from entity embeddings, relation embeddings, hierarchical attribute representations, and historical reasoning path states. Assuming that the number of entities and relations are denoted by $|E|$ and $|R|$, respectively, the overall embedding storage complexity can be expressed as:

$$O((|E| + |R| + N_a) \cdot d) \quad (33)$$

Although our framework introduces additional semantic interaction and feature fusion modules, the target-guided action pruning strategy effectively constrains redundant candidate exploration, thereby preventing excessive computational growth during multi-hop reasoning.

Overall, our framework maintains manageable computational complexity while preserving semantic interaction capability and reasoning flexibility, demonstrating favorable scalability for large-scale knowledge graph reasoning tasks.

4. Experiment

4.1. Datasets

The structural completeness and semantic consistency of datasets are essential for constructing high-quality knowledge graph reasoning models. Widely adopted benchmark datasets, such as WN18RR and FB15k-237, serve as standard evaluation platforms for multi-hop reasoning and knowledge graph completion tasks. WN18RR, derived from WordNet, emphasizes hierarchical lexical-semantic relations, including hypernymy and synonymy, and thus shows clear conceptual hierarchies and long-range semantic reasoning patterns. In contrast, FB15k-237, derived from Freebase, contains richer entity-relation interaction patterns, supporting evaluation of complex relational reasoning and multi-relational semantic inference. WN18RR comprises 40,943 entities and 11 relation types, with reasoning tasks focused on semantic hierarchy inference and long-range conceptual reasoning. FB15k-237 includes 14,541 entities and 237 relation types, emphasizing complex relational interactions and diverse semantic associations. Consequently, these two public datasets display complementary reasoning characteristics, specifically, WN18RR is more suitable for assessing conceptual hierarchy reasoning, while FB15k-237 better evaluates complex relational reasoning. Although WN18RR and FB15k-237 lack attribute semantic structures, their inclusion in experimental evaluations allows assessment of our framework's cross-domain generalization and reasoning effectiveness across diverse semantic structures.

The limitation of lacking fine-grained modeling limits the ability to evaluate attribute-aware reasoning frameworks in complex technical semantic environments. Therefore, we construct a Technical Knowledge Graph Dataset (TKGD) containing hierarchical attribute semantics and attribute-aware reasoning constraints to support multi-hop reasoning tasks under complex technical scenarios. The TKGD is constructed from publicly accessible web resources and technical documents, integrating entity information, relational semantics, and hierarchical attribute structures. To ensure semantic consistency and structural quality, ontology-guided semantic constraints together with semi-automatic extraction strategies are employed during dataset construction. Specifically, the dataset incorporates 89 entity categories, 297 attributes, and 73 semantic relation types, thereby forming a multi-level technical semantic graph with both relational and attribute-aware reasoning characteristics. In addition, semantic normalization and conflict resolution strategies are applied during dataset preprocessing to reduce semantic inconsistency and redundant entity representations. The final TKGD dataset contains 23,571 entities, 116,287 relational triples, and 241,632 attribute triples. The dataset exhibits a core-around structural distribution, where core entities are centered and expanded with associated attributes and related entities. To evaluate the generalization capability of models, the relational triples are split into training, validation, and test sets with a ratio of 8.5:0.75:0.75 as shown in Table 2. During the splitting process, it is enforced that the entity and relation types in the validation and test sets are subsets of those in the training set, thereby avoiding the cold-start problem and ensuring the validity of the evaluation.

Compared with public benchmark datasets such as WN18RR and FB15k-237, the TKGD contains richer attribute semantic information and hierarchical semantic depen-

dencies. Consequently, the dataset is more suitable for evaluating semantic interaction modeling, hierarchical attribute reasoning, and target-guided multi-hop reasoning under complex technical semantic environments.

Table 2. Statistics of the technical knowledge graph dataset (TKGD).

Data Split	Number of Entities	Number of Triplets	Percentage (%)
Training Set	23,571	98,844	85.0%
Validation Set	11,628	8721	7.5%
Test Set	11,655	8722	7.5%
Total	23,571	116,287	100.0%

4.2. Evaluation Metrics

To rigorously assess the performance of our knowledge graph reasoning model, we employ widely recognized evaluation metrics, including Mean Reciprocal Rank (MRR) and Hits@K for $K = 1, 3, 10$, which quantitatively capture our model's accuracy in ranking candidate triplets. Additionally, the Average Reasoning Steps metric is utilized to evaluate the efficiency of our model, particularly in optimizing reasoning paths and compressing the action space. Regarding ranking accuracy, MRR computes the average countdown rank of the correct target entities in our model's predicted rankings. Higher MRR values represent superior retrieval precision, with the formal definition as follows:

$$MRR = \frac{1}{|S|} \sum_{i=1}^{|S|} \frac{1}{rank_i}$$

where $|S|$ denotes the total number of triplets in the test set, and $rank_i$ represents the predicted rank of the correct target entity for the i -th test sample. Simultaneously, the Hits@K metric measures the proportion of samples where the correct target entity is included within the top K recommended results, reflecting our model's ability to rank correct answers at the forefront of the search results. We record the results for $K = 1, 3, 10$, with the specific definition as follows:

$$Hits@K = \frac{1}{|S|} \sum_{i=1}^{|S|} \mathbb{I}(rank_i \leq K) \times 100\%$$

where $\mathbb{I}(\cdot)$ is the indicator function, taking the value 1 if the predicted rank falls within the top K , and 0 otherwise. To further evaluate reasoning efficiency, Average Reasoning Steps measures the mean number of discrete actions required for our model to traverse from the head entity to the target entity, thereby reflecting our model's ability to suppress redundant paths and produce concise reasoning chains under the reinforcement learning framework, consistent with principles discussed in reinforcement learning for single-agent to multi-agent systems. The calculation is expressed as:

$$Avg_Steps = \frac{1}{|S|} \sum_{i=1}^{|S|} L_i$$

where L_i is the number of discrete action steps taken by the model when processing the i -th query task. By comprehensively analyzing both accuracy and efficiency metrics, the overall reasoning effectiveness of our model in processing complex semantic associations can be evaluated.

4.3. Experimental Setup

To ensure repeatability and enable fair comparison, all experiments are performed under a standardized computing environment. Detailed hardware and software configurations are presented in Table 3. The CPU used in experiments is Intel Xeon Gold Series Processor manufactured by Intel Corporation in Santa Clara, CA, USA, and the GPU is NVIDIA GeForce RTX 3090 manufactured by NVIDIA Corporation in Santa Clara, CA, USA. All baseline models and our model were trained and evaluated under identical physical devices and software dependencies to eliminate potential interference arising from variations in computational performance or framework versions.

Table 3. Configuration of experimental environment.

Environment Item	Specific Configuration
Operating System	Ubuntu 20.04 LTS
CPU	Intel Xeon Gold Series Processor (32 Cores)
GPU	NVIDIA RTX 3090 (24 GB)
Memory	64 GB DDR4 RAM
Development Environment	Python 3.8, PyTorch 1.12.1, CUDA 11.6
Batch Size	64
Optimizer	Adam

To ensure fairness and consistency in runtime evaluation, all models were trained and evaluated under identical hardware, software frameworks, and optimization settings. The batch size was uniformly set to 64, and all models followed the same data preprocessing and evaluation methods. Training time refers to the average time required for one complete training process under the same convergence standard. Inference latency was measured as the average reasoning time per query during testing. GPU memory consumption was recorded as the peak used GPU memory during training, and parameter scale statistics were computed based on the total number of trainable model parameters.

To improve the reliability and reproducibility of experimental evaluation, each experiment was run five times using different random seeds: 42, 123, 3407, 2025, and 4096. The final report results are taken as the average of multiple rounds. Standard deviation statistics were also computed to assess the robustness and convergence stability of the reasoning models under random reinforcement learning optimization.

Paired *t*-test analyses were performed between our framework and major baseline models, including MINERVA, AdaPath, CURL, and HMR, to determine whether observed performance improvements were statistically significant. The significance level was uniformly set to $p < 0.05$. The reason for using the paired *t*-test is that all models are evaluated on the same dataset, training settings, and evaluation metrics, which can reduce the interference caused by random optimization fluctuations while fairly comparing performance differences.

Confidence interval distributions and standard deviation variations were analyzed to assess the stability of reasoning performance between different experimental rounds. The results indicate that hierarchical semantic modeling, adaptive feature fusion, and target-guided reasoning mechanisms effectively relieve strategy fluctuation and noise exploration during reinforcement learning optimization, thereby improving the stability of the training process and enhancing the consistency of reasoning in multiple rounds of experiments.

To better evaluate the convergence behavior of our model during training, we used several monitoring methods. We check convergence through cumulative reward trends and policy gradient loss curves recorded across epochs. We adjusted learning rate, batch size, and reward normalization to balance optimization speed and stability. In particular, reward normalization was applied to scale multi-dimensional reward signals into a consistent

range, reducing gradient explosion or vanishing issues. Iteration numbers were determined based on preliminary convergence studies, ensuring sufficient exploration for training stability without overfitting. These monitoring strategies help us detect early unstable updates, fluctuating reward patterns, or ineffective path exploration, thereby enhancing experimental control, reproducibility, and overall reliability.

To ensure the stability and effectiveness of our framework, key hyperparameters for the adaptive gating mechanism, query-aware attention module, hierarchical semantic modeling module, and reinforcement learning optimization process were uniformly configured based on preliminary experiments and validation set optimization. All experiments follow the controlled variable strategy, adjusting only one parameter at a time while keeping others fixed. The learning rate was initialized at 1×10^{-4} to balance convergence speed and training stability. Candidate learning rates were selected from 1×10^{-5} , 5×10^{-5} , 1×10^{-4} , 5×10^{-4} . Excessively large learning rates caused unstable strategy updates and fluctuating reward convergence, while excessively small rates slowed model convergence.

The hidden semantic dimension d in the adaptive fusion layer and query-aware attention mechanism was set to 256 to maintain semantic consistency across feature interaction, strategy learning, and action evaluation stages. Embedding dimension and attention hidden size were jointly explored within 128, 256, 512 based on validation set performance and convergence behavior. Larger hidden dimensions enhanced semantic representation but introduced additional computational overhead and potential training instability, whereas smaller dimensions reduced semantic discrimination during multi-hop reasoning. The query-aware attention hidden size and the embedding dimension were consistent to ensure stable semantic interaction and feature alignment throughout reasoning optimization.

In the query-aware attention mechanism, the attention scaling factor $\sqrt{d}$ was applied to smooth the attention distribution and relieve gradient explosion resulting from overly concentrated attention weights in high-dimensional semantic spaces. The relevance weighting coefficient γ in Equation (16) was set to 0.3 after validation to balance semantic guidance and reinforcement learning policy exploration. Candidate values for γ were 0.1, 0.3, 0.5, 0.7. Excessively large γ caused the policy network to over-rely on semantic attention scores, reducing exploration diversity, whereas small γ diminished the effectiveness of target-guided action constraints.

In the adaptive gating mechanism, the fusion coefficient $g \in [0, 1]$ was dynamically produced by the sigmoid activation function to achieve normalized semantic weighting between entity representations and hierarchical attribute features. Hierarchical attribute settings were derived from macro-level guiding attributes and micro-level discriminative attributes extracted from the technical knowledge graph schema. Preliminary experiments demonstrated that dynamic gating substantially improved semantic separability and reasoning stability compared with fixed-weight fusion strategies.

Reward-related hyperparameters in the reinforcement learning optimization stage were systematically adjusted based on validation to ensure stable policy convergence and suppress noisy reasoning behaviors. Reward coefficients for accuracy reward, semantic evolution reward, and step penalty were jointly optimized following a controlled variable strategy. The reward weight search range was empirically set to 0.1, 0.3, 0.5, 0.7, 1.0, and penalty coefficients for path length constraints and invalid action punishment were explored within -1.5 , -1.0 , -0.5 , -0.1 according to reasoning convergence behavior and path quality statistics.

Final reward coefficient settings were determined through a comprehensive evaluation of cumulative reward convergence, reasoning accuracy, average path quality, and strategy fluctuation on the validation set. The results indicate that excessively high reward coefficients led to overfitting to local reasoning paths and unstable policy updates, whereas

overly small coefficients reduced the effectiveness of semantic guidance and reinforcement learning optimization. Likewise, excessively strong penalty constraints diminished exploration diversity and long-distance reasoning capability, while insufficient penalties failed to suppress redundant path exploration and noisy reasoning behaviors.

The reinforcement learning discount factor was incorporated to balance immediate rewards with long-term reasoning optimization. The discount factor η was empirically explored within 0.85, 0.90, 0.95, 0.99 and ultimately set to 0.95 based on cumulative reward convergence trends and reasoning stability. Larger discount factors improved long-horizon reasoning sensitivity but occasionally induced delayed reward fluctuation, whereas smaller discount factors overemphasized short-term gains and weakened multi-hop semantic dependency modeling.

Parameter sensitivity experiments were conducted to evaluate the robustness and statistical stability of our framework. The parameters examined included the learning rate, relevance weighting coefficient γ , hidden semantic dimension d , reward coefficients, penalty levels, discount factor η , and reasoning depth L . Reasoning depth was empirically explored within 2, 3, 4, 5 based on reasoning accuracy, convergence behavior, and computational efficiency. Each parameter sensitivity experiment was independently repeated under identical environments and random seed settings, varying only the target hyperparameter per experimental group to ensure fair comparison and controlled analysis.

Multiple optimization statistics are recorded during training to analyze convergence and evaluate strategy stability, including cumulative reward, training loss, average reasoning path length, Valid Action Ratio, and average path quality. Among them, the cumulative reward reflects average accumulated reward, the training loss was from policy gradient optimization results, and the average reasoning path length evaluates semantic consistency and reaching the target. Parameter sensitivity experiment results are obtained from the statistical average after multiple rounds. Experiments show our model usually gets better reasoning accuracy, a more stable convergence process, and higher reasoning efficiency within moderate parameters, thus verifying the rationality of parameter settings. The low standard deviation and stable cumulative reward convergence among different rounds show that our model has better robustness and stable convergence ability in different initializations and environments.

4.4. Comparative Experiments

Comparative experimental results on the TKGD dataset are summarized in Table 4. All results were calculated as averages across five independent rounds under different random seeds and are presented in the format of mean $\pm$ standard deviation. Paired t -tests with a significance threshold of $p < 0.05$ were conducted to assess statistical significance. Across all evaluation metrics, our model achieved state-of-the-art performance, with MRR of 0.639 ± 0.003 and *Hits@10* of $74.0 \pm 0.4\%$. In terms of methodology, our framework is different from control methods in other fields. The dynamic event-triggered control keeps system stability by reducing unnecessary system control updates in continuous-time systems. The adaptive dynamic programming finds the best control strategy in unknown system rules. Our method is used for knowledge graph multi-hop reasoning. The semantic structure is discrete and multi-level, directly influencing how to choose the action step by step. Our framework contains three key mechanisms. Firstly, a dynamic path constraint is used to limit the search in the related semantics range. Secondly, the action selection method is guided by the target, which makes the reasoning process trend the entity and relation related to the target. The third is adaptive usage of hierarchical attribute information, where the macro feature provides global semantic guidance and the micro feature captures differences in detail, helping our model to distinguish different entities and relationships.

These mechanisms improve the ability of our model to face specific tasks. At the same time, they enhance semantic interpretability and ensure efficiency. Therefore, our method is different from traditional control methods and is designed around relation semantic reasoning, having advantages in knowledge graph reasoning tasks.

Table 4. Comparative experimental results on the TKGD dataset.

Model	Hit@1 (%)	Hit@3 (%)	Hit@10 (%)	MRR
DeepPath	48.3 ± 0.7	55.6 ± 0.6	63.2 ± 0.6	0.537 ± 0.007
MINERVA	51.2 ± 0.6	58.4 ± 0.5	66.1 ± 0.5	0.564 ± 0.006
AdaPath	54.1 ± 0.5	61.3 ± 0.5	69.2 ± 0.4	0.593 ± 0.006
CURL	56.4 ± 0.4	63.6 ± 0.4	71.5 ± 0.4	0.615 ± 0.005
HMR	57.8 ± 0.4	64.9 ± 0.4	72.8 ± 0.3	0.628 ± 0.004
Ours	58.9 ± 0.3 *	66.1 ± 0.3 *	74.0 ± 0.4 *	0.639 ± 0.003 *

* indicates statistically significant improvement compared with the strongest baseline model under paired *t*-test ($p < 0.05$).

To further evaluate the cross-domain generalization capability of our framework, additional comparative experiments were conducted on the public benchmark datasets WN18RR and FB15k-237. The experimental results on WN18RR and FB15k-237 are summarized in Table 5 and Table 6, respectively.

Table 5. Comparative experimental results on the WN18RR dataset.

Model	Hit@1 (%)	Hit@3 (%)	Hit@10 (%)	MRR
DeepPath	18.2 ± 0.6	32.5 ± 0.5	48.7 ± 0.5	0.254 ± 0.006
MINERVA	20.5 ± 0.5	35.8 ± 0.5	52.3 ± 0.4	0.281 ± 0.005
AdaPath	21.6 ± 0.5	36.8 ± 0.4	53.7 ± 0.4	0.295 ± 0.005
CURL	22.9 ± 0.4	38.5 ± 0.4	55.6 ± 0.3	0.310 ± 0.004
HMR	23.3 ± 0.4	39.0 ± 0.3	56.3 ± 0.3	0.315 ± 0.004
KG-BERT	4.1 ± 0.2	30.2 ± 0.3	38.3 ± 0.3	0.216 ± 0.004
Ours	23.1 ± 0.3	38.5 ± 0.3	55.9 ± 0.3	0.312 ± 0.003

The superior performance on the TKGD dataset primarily stems from the collaborative effects of hierarchical semantic modeling, adaptive feature interaction, and target-guided policy optimization. Early reinforcement learning-based reasoning models, such as DeepPath and MINERVA, mainly focus on sequential relation path traversal, where policy learning relies on relation transition exploration while using semantic constraints from entity attributes. As a result, these methods show blind path exploration and error accumulation in TKGD, where attributes and relations are highly semantically correlated.

Table 6. Comparative experimental results on the FB15k-237 dataset.

Model	Hit@1 (%)	Hit@3 (%)	Hit@10 (%)	MRR
DeepPath	16.3 ± 0.6	30.1 ± 0.5	45.6 ± 0.5	0.228 ± 0.006
MINERVA	18.7 ± 0.5	33.4 ± 0.5	49.2 ± 0.4	0.253 ± 0.005
AdaPath	19.7 ± 0.4	34.4 ± 0.4	50.6 ± 0.4	0.267 ± 0.005
CURL	20.9 ± 0.4	35.9 ± 0.4	52.4 ± 0.3	0.282 ± 0.004
HMR	21.3 ± 0.3	36.5 ± 0.3	52.9 ± 0.3	0.286 ± 0.004
Ours	21.2 ± 0.3	36.2 ± 0.3	52.7 ± 0.3	0.284 ± 0.003

Subsequent studies improved reasoning capability from different perspectives. AdaPath enhances long-range dependency retention and reasoning adaptability by adaptive path

memory mechanisms; however, its optimization focuses on memory evolution rather than semantic hierarchy construction. CURL improves training stability and reduces reasoning redundancy through collaborative dual-agent optimization and action space exploration, but treats attributes as flattened semantic features during policy learning. HMR introduces hierarchical reasoning structures to decompose reasoning stages and improve efficiency; however, its hierarchy mainly reflects the reasoning process rather than the hierarchical semantic organization of entity attributes. Consequently, these methods lack the ability to differentiate macro-level guiding attributes from micro-level discriminative attributes and do not incorporate explicit target-oriented semantic constraints during action selection.

In contrast, our framework models hierarchical attribute semantics by distinguishing macro-level semantic guidance from micro-level discriminative constraints. The adaptive hierarchical feature fusion module dynamically coordinates semantic contributions based on the current reasoning state, historical reasoning paths, and target semantics, enhancing the semantic separability of reinforcement learning states and improving policy optimization stability. Although additional semantic interaction and hierarchical fusion modules are introduced, overall reasoning efficiency remains competitive due to target-guided action pruning and dynamic action selection mechanisms. By filtering low-relevance candidate actions according to target semantics, our framework effectively reduces redundant exploration and mitigates computational overhead commonly observed in large-scale multi-hop reasoning tasks.

Experimental results on public datasets further demonstrate the generalization capability of our framework. To assess runtime efficiency and scalability, additional analyses were conducted on training time, average inference latency, parameter scale, and GPU memory usage under identical hardware environments. Compared with traditional reinforcement learning-based reasoning models such as MINERVA and AdaPath, our framework introduces semantic interaction modules, including hierarchical semantic modeling and adaptive feature fusion, which slightly increase parameter scale and training overhead. Nevertheless, with target-guided action pruning and dynamic action selection mechanisms, our framework effectively reduces invalid path exploration and redundant action evaluations during multi-hop reasoning.

Experimental observations indicate that our model maintains competitive inference efficiency compared with advanced reasoning frameworks, including CURL and HMR, while achieving superior reasoning accuracy and more stable optimization. The target-guided action constraint mechanism substantially compresses the effective action search space, reducing reasoning delay and enhancing convergence efficiency in large-scale knowledge graph environments. Furthermore, compared with methods relying on exhaustive path exploration or extensive semantic interaction, our framework achieves a balanced trade-off between reasoning accuracy, computational overhead, and scalability.

Although WN18RR and FB15k-237 lack rich hierarchical attribute semantic information, our framework still achieves competitive reasoning performance compared with advanced reinforcement learning-based reasoning models. However, it should be emphasized that both WN18RR and FB15k-237 are primarily constructed from entity-relation-entity triples and do not provide sufficient hierarchical attribute descriptions for semantic attribute interaction modeling. Consequently, the hierarchical attribute modeling and adaptive feature fusion mechanisms proposed in our work cannot be fully exploited under such benchmark settings. In this scenario, the performance improvements over advanced models such as CURL and HMR become relatively limited on certain evaluation metrics, indicating that our framework exhibits a certain degree of dependence on attribute semantic information during multi-hop reasoning optimization.

On WN18RR, our model achieves an MRR of 0.312 and *Hits@10* of 55.9%, outperforming earlier reinforcement learning-based reasoning frameworks including DeepPath, MINERVA, and AdaPath, while remaining close to CURL and HMR. Since WN18RR mainly emphasizes conceptual hierarchy reasoning and long-distance relational dependency modeling, the target-guided action constraint mechanism effectively suppresses invalid path exploration and improves semantic consistency during sequential reasoning. Nevertheless, due to the absence of explicit hierarchical attribute semantics, the advantages of adaptive hierarchical semantic fusion cannot be comprehensively reflected in this dataset.

Similarly, on FB15k-237, our model achieves an MRR of 0.284 and *Hits@10* of 52.7%, remaining competitive with CURL and HMR. FB15k-237 contains more complex entity-relation interaction patterns and diversified relational semantics, which place higher demands on multi-hop path optimization and semantic-aware action selection. However, the dataset lacks fine-grained attribute semantic structures; thus, the macro-level semantic guidance and micro-level discriminative attribute modeling introduced in our framework cannot provide sufficient semantic constraints during reasoning optimization. Therefore, the comparative experimental results on public datasets mainly validate the general reasoning capability, optimization stability, and action decision effectiveness of our framework in generic knowledge graph environments, rather than fully demonstrating the advantages of hierarchical attribute modeling and adaptive semantic fusion. These observations further indicate that the cross-domain generalization capability of our framework under attribute-deficient scenarios still exhibits certain limitations and requires further investigation in future research.

Methodological Mechanism Comparison and Discussion. Existing knowledge graph completion approaches can be broadly classified into several paradigms, including GNN-based methods such as CompGCN, NBFNet, and GraIL, as well as pre-trained Transformer-based models exemplified by KG-BERT. Including KG-BERT in the comparison helps improve experiment evaluation integrity and rigor. The experiment results show that our model has multiple higher indicators compared with early reinforcement learning methods on WN18RR and FB15k-237. KG-BERT, which only relies on the text encoding capability of Transformer, cannot deal with complex multi-hop relation reasoning, thus performing weakly in link prediction. Further comparison with KG-BERT shows the advantages of our work. Firstly, it enriches the evaluation of different technical routes, indicating that our method can keep a stable multi-hop reasoning ability on different datasets that lack clear attribute semantics. Secondly, the comparison shows the advantages of our model in reinforcement learning reasoning under complex relationship constraints.

CNNs are better at dealing with local information and capturing a graph structure model, but they lack reasoning path explanations. Transformers focus on semantic representation and the usage of language knowledge, but they regard link prediction as a classification problem. Our method combines dynamic path search, target guidance, and reasoning path together, thereby balancing reasoning effectiveness, interpretability, and efficiency.

4.5. Ablation Study

To evaluate the specific contributions of the key components within our model to its reasoning, we conducted completion performance, ablation experiments, reward mechanism analyses, and parameter sensitivity experiments on the TKG dataset. By constructing and comparing model variants with missing mechanisms, we provide an in-depth analysis of the roles played by hierarchical attribute modeling, adaptive fusion strategies, goal-guided constraints, and path-aware rewards within the overall architecture. In addition, sensitivity experiments were further conducted for reward coefficients, penalty levels,

embedding dimensions, attention configurations, reasoning depth, and hierarchical attribute settings to comprehensively evaluate the robustness, convergence stability, and collaborative effectiveness of different architectural components.

To further analyze the optimization behavior of our reinforcement learning framework during training, cumulative reward curves, training convergence curves, and strategy stability statistics were additionally recorded throughout the learning process. Specifically, the cumulative reward curve was used to measure the long-term reward growth trend and reasoning policy improvement capability during training, while the convergence curve recorded the variation in reinforcement learning loss values and reasoning accuracy under different training epochs. In addition, policy stability analysis was conducted by statistically evaluating the variance of reasoning rewards and action selection distributions across multiple training rounds under different random seeds. All variants and parameter configurations were evaluated under a unified data partition, training protocol, and evaluation setting to ensure fair comparison and experimental reproducibility.

The ablation study analyzes the convergence and stability of different model variants in the training process. Through cumulative reward and strategy gradient loss curve, we find that the lack of reward shaping results in slower reward growth, the lack of adaptive hierarchical fusion leads to more unstable actions, and the lack of target guidance action constraints results in longer convergence times. These results further show that reward shaping, adaptive hierarchical fusion, and goal-guided action constraints can work together to make our model more stable and reduce ineffective path exploration during training.

Regarding the construction of model variants, *w/o* Hierarchical Attributes removes the attribute hierarchy mechanism and treats entity attributes as flattened, unified vector representations to verify the necessity of hierarchical modeling in capturing macro-guiding and micro-matching features; *w/o* Adaptive Fusion eliminates the adaptive gated fusion mechanism and employs fixed weights for merging hierarchical attributes to assess the effectiveness of dynamic weight allocation in meeting attribute requirements at different reasoning stages; *w/o* Target-guided Action Constraint removes the goal-based action space constraint, making action selection solely dependent on the current state to validate the core role of target semantic information in compressing redundant action spaces; *w/o* Path-aware Reward simplifies the reward function by retaining only the terminal task reward and removing intermediate rewards related to path quality, aimed at measuring the impact of the reward mechanism on the logical coherence of reasoning chains.

The hierarchical attribute division mechanism, implemented via semantic anchors, plays a critical role in distinguishing coarse-grained and fine-grained attribute information. By separately modeling macro semantic guidance information and micro features, our model can better use attribute information related to the current task in the multi-hop reasoning process. Based on this mechanism, the weighted scoring mechanism further evaluates the importance of different attributes, making our model focus on the more key attributes, and reducing the impact of low-value attributes. The combination of the two mechanisms improves the efficiency of feature usage and makes action selection more stable, thereby ensuring the consistency of the reasoning strategy in the long path reasoning process. In the ablation experiment, the performance reduction in the *w/o* Hierarchical Attributes variant, especially the decrease in MRR index, confirmed the necessity of the above mechanism at the experimental level. The experimental results of the *w/o* Adaptive Fusion variant show that if the same processing method is applied to features at different levels, it will weaken decision efficiency and reduce overall reasoning performance. These experimental results show the rationality and effectiveness of the hierarchical attribute partitioning and weighted scoring mechanism in our framework.

Furthermore, reward sensitivity experiments are conducted. Under the principle of controlling variables and keeping other parameters unchanged, we adjust the reward coefficient and penalty levels. The reward coefficient search range was configured within $\{0.1, 0.3, 0.5, 0.7, 1.0\}$, while the penalty level range was explored within $\{-0.2, -0.5, -1.0, -1.5\}$. Additional parameter sensitivity experiments were also conducted for embedding dimensions, attention hidden sizes, reasoning depths, and hierarchical attribute configurations. The results of the ablation study are summarized in Table 7. The experimental results show that the complete model is better than each variant in all evaluation indicators, verifying the science of each functional module and the advantages of module cooperation. At the same time, additional parameter sensitivity experiments and reward stability experiments further show that our framework can maintain relatively stable reasoning performance and convergence performance in different parameters, thus verifying that the multimodule cooperation has good robustness and generalization ability.

Table 7. Ablation study results on the TKGD dataset.

Variant	Hit@1 (%)	Hit@3 (%)	Hit@10 (%)	MRR
<i>w/o</i> Hierarchical Attributes	53.2	60.5	68.8	0.581
<i>w/o</i> Adaptive Fusion	55.5	63.2	71.5	0.606
<i>w/o</i> Target-guided Action Constraint	52.1	59.3	67.6	0.570
<i>w/o</i> Path-aware Reward	57.2	64.8	72.9	0.624
Full Model (Ours)	58.9	66.1	74.0	0.639

The analysis indicates that target-guided action constraint and hierarchical attribute fusion are the core components supporting high-performance reasoning. The additional sensitivity experiments further reveal that different parameter settings have varying influences on semantic representation quality, reasoning stability, reward convergence behavior, and action selection effectiveness. The performance of the *w/o* Target-guided Action Constraint variant dropped most severely, with its MRR decreasing by 0.069 and *Hits*@1 falling to 52.1%. This is attributed to the loss of target semantic guidance, which prevents our model from effectively filtering weakly correlated actions, leading to an exploded action space and paths that easily deviate from the target direction.

The reward sensitivity analysis further demonstrates that moderate reward coefficients significantly improve reasoning stability, cumulative reward growth rate, and target-oriented path exploration capability, whereas large reward coefficients tend to cause policy overfitting to local reasoning trajectories and unstable optimization behavior. In contrast, small reward coefficients weaken the effectiveness of semantic guidance and slow the convergence speed of policy optimization. Additional penalty-level experiments further indicate that appropriate penalty constraints effectively suppress redundant path exploration and reduce invalid reasoning behaviors, while overly strong penalties negatively affect exploration diversity and cause premature convergence.

The cumulative reward curves and training convergence curves show that our framework can more quickly accumulate reward, compared with the simplified reward mechanism model. In the early training stage, cumulative reward steadily increased with small fluctuation, showing that multiple rewards can relieve the sparse reward problem and improve strategy learning efficiency. As the training progresses, the convergence curve is gradually stabilizes, and the reinforcement learning loss fluctuates less, showing that the reward balance strategy helps reduce the strong fluctuation and noise in training. We conduct multiple rounds of experiments based on multiple random seeds, and we find that our framework can keep a consistent reward distribution and action selection probability. Balanced reward mechanism effectively reduces the variance of rewards and improves

strategy robustness in the reasoning process, compared with the simplified reward mechanism model. Overall, the combination of reward normalization, path perception rewards, and target guidance penalty constraints effectively enhances reasoning consistency and training stability.

Experiments with different attention hidden sizes indicate that moderate attention configurations consistently achieve better semantic filtering capability and action relevance estimation, while excessively small hidden sizes weaken semantic discrimination and excessively large configurations introduce redundant computation and unstable attention distributions. The MRR of the *w/o* Hierarchical Attributes variant decreased by 0.058, proving that failing to distinguish the semantic weights of macro and micro attributes hinders our model's ability to capture the deep coupling between attributes and relations. Additional hierarchical attribute configuration experiments further indicate that the collaborative modeling of coarse-grained semantic guidance and fine-grained discriminative semantics significantly improves reasoning consistency and semantic separability compared with flattened attribute representations.

In contrast, the adaptive fusion module and path-aware reward module serve as critical auxiliary optimizations. Experiments with different embedding dimensions additionally demonstrate that moderate semantic dimensions achieve a favorable balance between representation capability and computational complexity, while excessively small dimensions weaken semantic interaction capability and excessively large dimensions introduce unnecessary computational overhead and training instability. The performance decline of *w/o* Adaptive Fusion suggests that fixed weights limit the flexibility of attribute invocation, while the impact of *w/o* Path-aware Reward was relatively minor, as evidenced by a decrease of only 0.015 in MRR, indicating that the core reasoning architecture remains robust even under a simplified reward mechanism.

Reasoning depth sensitivity analysis shows that shallow reasoning depth can limit the ability of model semantics reasoning over long distances, while deep reasoning depth can cause continuously increasing errors and excess exploration. Moderate reasoning depth can get better reasoning accuracy and convergence stability. Experiments prove that our framework enhances knowledge graph reasoning accuracy and efficiency. The performance of our model under different parameters is stable, which verifies our framework's robustness, extensibility, and convergence stability.

4.6. Reasoning Efficiency and Search Behavior Analysis

In practical applications of knowledge graph reasoning, reasoning efficiency and reasoning accuracy are of equal importance. Efficient reasoning behavior can significantly improve semantic consistency and path rationality during multi-hop reasoning processes. To verify the advantages of our model in terms of reasoning efficiency, experiments are conducted on representative reasoning scenarios within a technical knowledge graph dataset. A statistical analysis is performed on several metrics across different models, including Average Reasoning Steps, Valid Action Ratio, and Reasonable Path Ratio. The Valid Action Ratio is defined as the proportion of actions highly relevant to the target among all candidate actions, while the Reasonable Path Ratio refers to the proportion of generated reasoning paths that can effectively lead to the target entity among all sampled paths. All reported results are averaged over multiple rounds, as shown in Figure 2.

As illustrated in Figure 2, our model consistently outperforms baseline methods across all three metrics. In terms of reasoning steps, our model reduces the average number of steps to 2.1, achieving a reduction of 0.7 steps compared with MINERVA and 0.2 steps compared with HMR. This improvement is mainly attributed to the target-guided action constraint module, which dynamically compresses the candidate action space by

accurately selecting actions strongly associated with the target entity. This effectively reduces redundant action exploration and enables the reasoning process to converge to the target entity in fewer steps. Such streamlined search trajectories are particularly beneficial for maintaining semantic focus and reducing reasoning deviation during long-horizon reasoning processes. In terms of action selection effectiveness, our model achieves a Valid Action Ratio of 76.0 percent, outperforming HMR by 4.5 percentage points. This improvement stems from the fine-grained filtering capability of the target-aware attention mechanism, which identifies strongly relevant semantic actions while suppressing weakly related ones, thereby improving semantic relevance consistency during action selection and enhancing reasoning interpretability.

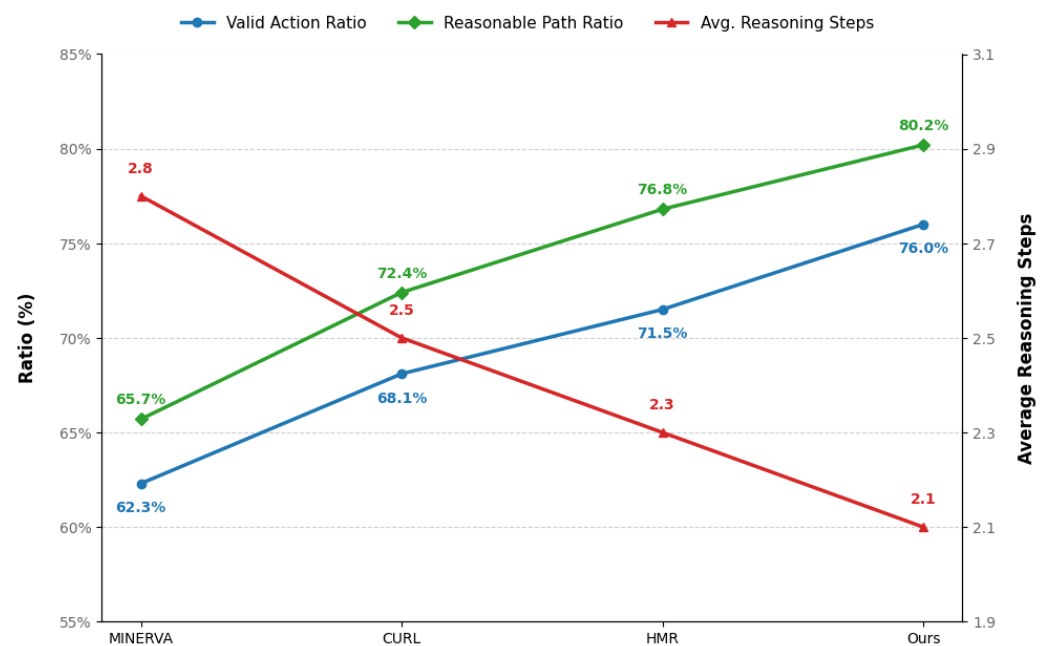


Figure 2. Comparison of reasoning efficiency and path rationality

Regarding path quality, our model achieves a Reasonable Path Ratio of 80.2 percent, surpassing HMR by 3.4 percentage points. This improvement is mainly attributed to the synergistic effect of the hierarchical attribute fusion module and the target-guided action constraint module. The former provides fine-grained attribute constraints to prevent path deviation caused by insufficient attribute utilization, while the latter offers explicit target guidance to reduce blind exploration. Their combination effectively suppresses the generation of invalid paths and enhances the logical consistency of the overall reasoning process.

In summary, our model demonstrates superior reasoning path quality and semantic-guided exploration capability in multi-hop knowledge graph reasoning. Importantly, this improvement is achieved without sacrificing reasoning accuracy, thereby realizing a balanced optimization between reasoning efficiency and path quality, which fully reflects the effectiveness of our model design and its practical applicability in complex semantic environments.

5. Conclusions

Our work addresses the core challenges in knowledge graph reasoning, including insufficient feature utilization and low reasoning efficiency, and proposes a multi-level reinforcement learning reasoning model based on adaptive hierarchical feature fusion. Our model achieves simultaneous improvements in reasoning accuracy and search efficiency through the coordinated design of multi-dimensional state representation, adaptive hierarchical feature fusion, action decision-making, and a multi-dimensional reward feedback

mechanism. Besides, our framework also shows promising practical application potential in real-world intelligent knowledge systems. In industrial knowledge management scenarios, the hierarchical semantic modeling mechanism can improve semantic organization capability and fine-grained knowledge representation for complex technical entities and heterogeneous attribute structures. In technical maintenance reasoning environments, our framework can support fault association analysis and hidden dependency discovery among equipment states, operational records, and maintenance events through target-guided semantic path exploration. Furthermore, in intelligent recommendation systems and question answering applications, the adaptive feature fusion and target-guided reasoning mechanisms can effectively improve semantic retrieval capability, reasoning response quality, and decision efficiency under large-scale dynamic knowledge environments. Therefore, our framework possesses strong application potential and practical promotion value in complex domain-oriented knowledge systems.

6. Limitations and Future Work

In practical applications of knowledge graph reasoning, reasoning efficiency and reasoning accuracy are of equal importance. Efficient reasoning capability can significantly reduce unnecessary search space exploration and improve system response speed. Although our method achieves notable progress in knowledge graph reasoning tasks, several limitations remain. First, existing studies primarily rely on structured knowledge entries for reasoning, while the utilization of unstructured data widely present in real-world scenarios remains insufficient. Future work may incorporate multimodal data such as images and technical documents to construct multimodal feature fusion-based reasoning models, aiming to uncover complex latent relationships within unstructured data and further improve reasoning accuracy.

Second, although our model demonstrates stable performance on specific domain datasets, its knowledge density remains insufficient when dealing with long-tail reasoning tasks characterized by scarce samples. Exploring cross-domain integration between domain-specific knowledge and general knowledge represents an important research direction. Domain adaptation techniques can be employed to reduce distributional discrepancies between different knowledge sources, thereby leveraging rich semantic information from general domains to enhance reasoning performance on sparse nodes.

From the perspective of modeling depth, although hierarchical attribute constraints have been incorporated into the reasoning process, the modeling of dynamic logical chains in entity evolution remains insufficient. Future research may consider introducing more sophisticated spatiotemporal evolution modeling mechanisms to capture dynamic relationships across the full lifecycle of entities. In addition, although our model guides a coarse-to-fine reasoning strategy through a multi-dimensional reward function, the computational overhead of structured reasoning modules remains a potential bottleneck in large-scale real-time deployment scenarios. Investigating lightweight reasoning architectures or knowledge distillation techniques to enable efficient deployment in edge computing environments will be a key direction for future research.

Author Contributions: Conceptualization, S.Z., C.H. and L.M.; methodology, S.Z., C.H., Y.W. and Q.T.; software, S.Z., C.H., Y.W. and J.Z.; validation, S.Z., C.H., Y.W. and X.L.; formal analysis, S.Z., C.H., Q.T. and L.W.; investigation, C.H., Y.W., Q.T. and Y.Y.; resources, L.M. and X.L.; data curation, S.Z., C.H., Y.W., Q.T. and J.Z.; writing—original draft preparation, S.Z. and C.H.; writing—review and editing, S.Z., C.H., L.M., X.L. and W.G.; visualization, J.Z., Y.Y. and W.G.; supervision, L.M. and X.L.; project administration, L.M. and X.L.; funding acquisition, L.M. and X.L.; All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by National Key Laboratory of Complex Aviation System Simulation Open Fund (SYS-KFSJJ-2025-1-07) and Young Elite Scientists Sponsorship Program of the Beijing High Innovation Plan (No. 20251027).

Data Availability Statement: The Technical Knowledge Graph Dataset (TKGD) constructed in this study is generated through the collection, curation, and structured annotation of open-source information from multiple publicly available channels. During the construction process, the original multi-source data undergoes fusion, cleaning, and semantic enhancement, resulting in an augmented knowledge graph dataset with high structural density and rich semantic value. In this process, the data may involve the reorganization and reconstruction of dispersed open-source information, and its redistribution may raise concerns regarding data integrity and usage boundary consistency. Moreover, certain data sources are subject to the licensing terms and non-commercial usage restrictions imposed by their original publishing platforms. Therefore, to avoid potential compliance risks and inconsistencies in data redistribution, the dataset is not publicly released.

Conflicts of Interest: Author Cheng He was employed by National Key Laboratory of Complex Aeronautical System Simulation, Beijing, China and author Liang Wang was employed by Shaanxi Aerospace Technology Application Research Institute Co., Ltd., Xi'an, China. The funder was not involved in the study design, collection, analysis, interpretation of data, the writing of this article, or the decision to submit it for publication. The remaining authors declare no conflict of interest.

References

- Hilario, M.; Nguyen, P.; Do, H.; Woznica, A.; Kalousis, A. Ontology-Based Meta-Mining of Knowledge Discovery Workflows. In *Meta-Learning in Computational Intelligence*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 273–315.
- Wang, W.Y.; Mazaitis, K.; Cohen, W.W. Programming with Personalized PageRank: A Locally Groundable First-Order Probabilistic Logic. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*; ACM, Inc.: New York, NY, USA, 2013; pp. 2129–2138.
- Gong, Z. Research on Ontology Reasoning Based on OWL Description. Ph.D. Thesis, Jilin University, Changchun, China, 2007.
- Song, H.; Zhao, G.; Sun, R. A Survey on Knowledge Graph Reasoning Based on Deep Reinforcement Learning. *Comput. Eng. Appl.* **2022**, *58*, 1–13.
- Yang, X. Research on Reasoning and Completion Algorithms Based on Knowledge Graph Construction. Master's Thesis, University of Electronic Science and Technology of China, Chengdu, China, 2020. <https://doi.org/10.27005/d.cnki.gdzku.2020.005071>.
- Chen, Y.; Goldberg, S.; Wang, D.Z.; Johri, S.S. Ontological Pathfinding. In *Proceedings of the 2016 ACM International Conference on Management of Data (SIGMOD)*; ACM, Inc.: New York, NY, USA, 2016; pp. 835–846.
- Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; Yakhnenko, O. Translating Embeddings for Modeling Multi-Relational Data. *Adv. Neural Inf. Process. Syst.* **2013**, *26*.
- Sun, Z.; Deng, Z.H.; Nie, J.Y.; Tang, J. RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. In *Proceedings of the International Conference on Learning Representations (ICLR)*, New Orleans, LA, USA, 6–9 May 2019.
- Li, D.; Liang, Y.; Qian, S.; Wu, H.; Jia, W.; Fu, Y.; Sun, Y. A Review of Background, Methods, Limitations and Opportunities of Knowledge Graph Completion. *Comput. Sci. Rev.* **2025**, *58*, 100809.
- Gallardo-Delgado, A.A.; Nolasco-Flores, J.A.; Priego-Sanchez, A.B.; Buscaldi, D. Large Language Models and Knowledge Graphs: A State-of-the-Art Exploration. *SN Comput. Sci.* **2025**, *6*, 752.
- Liu, S.; Li, W.; Liu, X.; Zhang, Y. A Survey on Knowledge Graph Reasoning Based on Reinforcement Learning. *Appl. Res. Comput.* **2024**, *41*, 2561–2572.
- Hou, Z.; Jin, X.; Chen, J.; Guan, S.; Wang, Y.; Cheng, X. A Survey on Explainable Reasoning over Knowledge Graphs. *J. Softw.* **2021**, *33*, 4644–4667.
- Tang, L.; Niu, Y.; Wang, R.; Xing, B.; Wang, Y. A Classification Study of Explainable Methods in Reinforcement Learning. *Appl. Res. Comput.* **2024**, *41*, 1601–1609.
- Xiong, W.; Hoang, T.; Wang, W.Y. DeepPath: A Reinforcement Learning Method for Knowledge Graph Reasoning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Copenhagen, Denmark, 7–11 September 2017; pp. 564–573.
- Kaelbling, L.P.; Littman, M.L.; Cassandra, A.R. Planning and Acting in Partially Observable Stochastic Domains. *Artif. Intell.* **1998**, *101*, 99–134.

16. Das, R.; Dhuliawala, S.; Zaheer, M.; Vilnis, L.; Durugkar, I.; Krishnamurthy, A.; Smola, A.; McCallum, A. Go for a Walk and Arrive at the Answer: Reasoning over Paths in Knowledge Bases Using Reinforcement Learning. In Proceedings of the International Conference on Learning Representations (ICLR), Vancouver, BC, Canada, 30 April–3 May 2018.
17. Schmidhuber, J.; Hochreiter, S. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780.
18. Shen, Y.; Chen, J.; Huang, P.S.; Guo, Y.; Gao, J. M-Walk: Learning to Walk over Graphs Using Monte Carlo Tree Search. *Adv. Neural Inf. Process. Syst.* **2018**, *31*.
19. Gedda, M.; Lagerkvist, M.Z.; Butler, M. Monte Carlo Methods for the Game Kingdomino. In Proceedings of the IEEE Conference on Computational Intelligence and Games (CIG); IEEE: Piscataway, NJ, USA, 2018. <https://doi.org/10.1109/CIG.2018.8490419>.
20. Zhang, D.; Yuan, Q.; Meng, L.; Xia, R.; Liu, W.; Qin, C. Reinforcement Learning for Single-Agent to Multi-Agent Systems: From Basic Theory to Industrial Application Progress, A Survey. *Artif. Intell. Rev.* **2026**, *59*, 46.
21. Liang, L.; Xiang, P.; Huang, H.; Zhang, N.; Qi, P.; Yin, Z.; Zhai, W.; Zhang, D. Sec-GNN-Driven Joint Multidimensional Anti-Eavesdropping Optimization for Secure UAV-Satellite Communications. *IEEE Internet Things J.* **2026**, *in press*.
22. Li, Y.; Zhang, H.; Wang, X.; Liu, Q. Dynamic Event-Triggered Control for Human-Machine Cooperative Systems Based on Dynamic Authority Allocation. *IEEE Trans. Ind. Inform.* **2024**, *20*, 8452–8464.
23. Chen, L.; Zhao, Y.; Sun, J.; Wu, M. Dynamic Event-Triggered Approximate Optimal Consensus Control for Unknown Nonlinear Multi-Agent Systems via Adaptive Dynamic Programming. *Inf. Sci.* **2024**, *675*, 120894.
24. Dong, H.; Ning, Z.; Wang, P.; Li, Y.; Shi, B. Adaptive Path-Memory Network for Temporal Knowledge Graph Reasoning. *arXiv* **2023**, arXiv:2304.12604.
25. Zhang, D.; Yuan, Z.; Liu, H.; Xiong, H. Learning to Walk with Dual Agents for Knowledge Graph Reasoning. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 22 February–1 March 2022; Volume 36, pp. 5932–5941.
26. Wang, Z.; Li, L.; Zeng, D.D. Hierarchical Multihop Reasoning on Knowledge Graphs. *IEEE Intell. Syst.* **2021**, *37*, 71–78.
27. Zheng, S.; Nie, X.; Yuan, W.; Qu, L.; Kong, X.; Zhang, Y.; Hou, J. Context-Aware Adaptive Reinforcement Learning for Multi-Hop Knowledge Graph Reasoning in Few-Shot Scenarios. *Knowl.-Based Syst.* **2025**, *331*, 114799.
28. Lin, X.V.; Socher, R.; Xiong, C. Multi-Hop Knowledge Graph Reasoning with Reward Shaping. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP), Brussels, Belgium, 31 October–4 November 2018; pp. 3243–3253.
29. Zheng, S.; Wang, W.; Qu, J.; Yin, H.; Chen, W.; Zhao, L. MMKGR: Multi-Hop Multi-Modal Knowledge Graph Reasoning. In 2023 IEEE 39th International Conference on Data Engineering (ICDE); IEEE: Piscataway, NJ, USA, 2023; pp. 96–109.
30. Saxena, A.; Tripathi, A.; Talukdar, P. Improving Multi-Hop Question Answering over Knowledge Graphs Using Knowledge Base Embeddings. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Virtual, 5–10 July 2020; pp. 4498–4507.
31. Wan, G.; Pan, S.; Gong, C.; Zhou, C.; Haffari, G. Reasoning like Human: Hierarchical Reinforcement Learning for Knowledge Graph Reasoning. In Proceedings of the International Joint Conference on Artificial Intelligence, Montreal, QC, Canada, 21–26 August 2021.
32. Zhu, A.; Ouyang, D.; Liang, S.; Shao, J. Step by Step: A Hierarchical Framework for Multi-Hop Knowledge Graph Reasoning with Reinforcement Learning. *Knowl.-Based Syst.* **2022**, *248*, 108843.
33. Vashishth, S.; Sanyal, S.; Nitin, V.; Talukdar, P. Composition-based Multi-Relational Graph Convolutional Networks. In Proceedings of the International Conference on Learning Representations (ICLR), Addis Ababa, Ethiopia, 26–30 April 2020.
34. Zhu, Z.; Zhang, Z.; Xhonneux, L.P.; Tang, J. Neural Bellman-Ford Networks: A General Graph Neural Network Framework for Link Prediction. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 29476–29487.
35. Teru, K.; Gaffney, E.; Hamilton, W. L. Inductive Relation Prediction by Subgraph Reasoning. In Proceedings of the International Conference on Learning Representations (ICLR), Addis Ababa, Ethiopia, 26–30 April 2020.
36. Yao, L.; Mao, C.; Luo, Y. KG-BERT: BERT for Knowledge Graph Completion. *arXiv* **2019**, arXiv:1909.03193.
37. Peng, J.; Wang, Z.; Li, Y.; Hou, P. Research on Multimodal Knowledge Graph Technology in Equipment Domain. *J. Ordnance Equip. Eng.* **2022**, *43*, 153.
38. Du, H.; Wang, B.; Han, J.; Bu, Z.; Cai, Y. Construction Strategies and Development Analysis of Knowledge Graphs for Aerospace Equipment. *Adv. Comput. Auton. Intell. Res.* **2024**, *2*, 118–121.
39. Liu, C.; Li, X.; Yu, Y.; Sun, Y. Research on Construction Method of Equipment Support Verification Knowledge Graph. *Software Engineering* **2020**, *23*, 5–8. *Softw. Eng.* **2020**, *23*, 5–8.
40. Liu, C.; Kong, X.; Wang, X.; Chen, W.; Li, H. Construction and Reasoning Method of Equipment Maintenance Support Knowledge Model. *Fire Control Command Control* **2024**, *49*, 151–160.
41. Yu, D.; Su, C.; Tian, H.; Song, W.; Yue, Y.; Bao, H. Construction and Reasoning Method of Knowledge Graph for Aircraft Skin Spraying Process. *Processes* **2026**, *14*, 581.

42. Qi, X.; Yang, B.; Wang, S.; Zhang, Z.; Zhang, Y.; Du, K. Few-Shot and Chain-of-Thought Prompting for Equipment Maintenance Knowledge Graph Construction via Large Language Models. *Knowl.-Based Syst.* **2026**, *335*, 115266.
43. Wang, C.; Mandelli, D.; Godbole, C.M.; Agarwal, V.; Movassat, M.; Mori, B.; Liang, D.; Nur, E.; Birjandi, A.; Lobo, B.; et al. Fusion of Equipment Reliability Data through Knowledge Graphs. *Reliab. Eng. Syst. Saf.* **2025**, *267*, 111785.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.