SCE: Semantic Consistency Enhanced Reinforcement Learning for Multi-Hop Knowledge Graph Reasoning

Yanwen Huang[1], **Yao Liu**[1,*], **Qiao Liu**[1], **Rui Hou**[1], **Tingting Dai**[1]
[1] University of Electronic Science and Technology of China
{huangyw@uestc.std.edu.cn, liuyao@uestc.edu.cn}

Abstract

Multi-hop reasoning with reinforcement learning has proven effective in discovering inference paths in incomplete knowledge graphs. However, a major challenge remains: spurious paths (incorrect reasoning paths that accidentally lead to correct answers) often arise due to reward mechanisms that prioritize final results over reasoning quality. While existing approaches attempt to mitigate this issue using external rules, they often neglect the internal semantic consistency between the target triple and the intermediate triples along the reasoning path. In this paper, we propose a novel framework, **S**emantic **C**onsistency **E**nhanced Reinforcement Learning (SCE), which incorporates semantic consistency into the reward function to guide multi-hop reasoning. Experimental results demonstrate that SCE outperforms strong baseline methods and facilitates the discovery of more interpretable reasoning paths. [1]

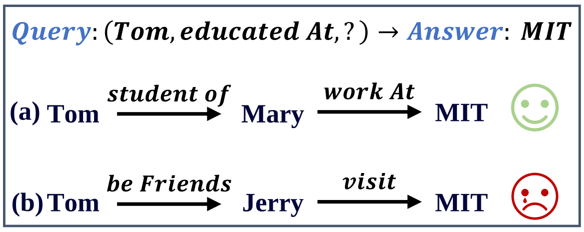

Figure 1: Two reasoning paths for the query; (a) is a reasonable path, while (b) is a spurious path.

1 Introduction

Knowledge Graphs (KGs) are structured collections of factual triples (Kok and Domingos, 2007; Dettmers et al., 2018; Chen et al., 2020) that have been widely applied in downstream natural language processing (NLP) tasks, such as recommendation systems (Guo et al., 2020; Wang et al., 2019; Cao et al., 2019) and question answering (Yasunaga et al., 2021; Huang et al., 2019). However, the inherent incompleteness of most KGs, due to missing facts, limits their effectiveness in these applications (Ji et al., 2021). Knowledge Graph Reasoning (KGR) has emerged as a promising solution to infer missing information.

Embedding-based methods (Bordes et al., 2013; Dettmers et al., 2018) aim to learn vector representations of entities and relations, capturing implicit relationships in high-dimensional spaces. While these approaches yield strong performance in link prediction tasks, they lack interpretability, which is crucial for understanding model decisions (Hildebrandt et al., 2020). To address this issue, rule-based methods (Yang et al., 2017; Cheng et al., 2022) have been proposed to mine logical rules (Guo et al., 2016; Meilicke et al., 2018), offering improved explainability. However, they often suffer from poor generalization and high computational cost (Zhang et al., 2022b).

Multi-hop Reasoning Methods (MRMs) (Lin et al., 2018; Wan and Du, 2021; Zhang et al., 2022a; Drance et al., 2023; Zhu et al., 2023), which incorporate reinforcement learning (RL), strike a balance between performance and interpretability. These methods model knowledge inference as a sequential decision-making process. For instance, given a query *(Tom, educatedAt, ?)*, a reasoning agent may correctly reach the answer *MIT* via a semantically coherent path (a) in Figure 1. However, the same answer could also be reached via an unrelated path (b), if the reward is based solely on the final answer. This issue—where incorrect reasoning leads to correct answers—is known as the spurious path problem, a persistent challenge in MRMs (Lin et al., 2018; Lv et al., 2021). This stems from reward functions that focus on factual accuracy while overlooking the semantic plausibility of the reasoning path (Das et al., 2017).

Several methods have been proposed to mitigate this issue, including action dropout (Lin et al.,

* Corresponding author
[1] Our code is available at uestc-huangyw/SCE

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 5413–5425
November 4-9, 2025

2018), high-quality rule mining (Lei et al., 2020), and spuriousness metrics (Jiang et al., 2023). However, these approaches typically rely on external rules and fail to leverage the internal semantic consistency among triples within a reasoning path.

In this work, we propose a novel solution centered on enhancing the semantic consistency of reasoning paths. As illustrated in Figure 1, the triples in path (a)—(Tom, student of, Mary) and (Mary, works at, MIT)—are more semantically aligned with the target triple (Tom, educatedAt, MIT) than those in path (b)—(Tom, is friends with, Jerry) and (Jerry, visits, MIT). This semantic consistency is influenced by both the relations and the entities involved. We define semantic consistency functions to measure such consistency and incorporate them into the learning process. Our contributions are summarized as follows:

1. We design relation and entity semantic consistency functions to quantify the consistency between intermediate and target triples.
2. We integrate the semantic consistency score into the reward function, enabling the agent to prioritize more interpretable reasoning paths.
3. Experimental results demonstrate that our method not only surpasses strong baselines in performance but also significantly improves interpretability and training efficiency.

2 Related Work

Embedding-based methods such as TransE (Bordes et al., 2013), TransR (Lin et al., 2015), and ComplEx (Trouillon et al., 2016) have been widely adopted due to their efficiency and strong predictive performance. These methods predict missing triples in KGs by embedding entities and relations into low-dimensional vector spaces, utilizing similarity or distance between embedding vectors. However, the lack of interpretability in these approaches hinders their practical applications.

To address this limitation, several hybrid approaches have emerged that aim to combine the strengths of embedding-based models with interpretable reasoning techniques. For instance, RLogic (Cheng et al., 2022) learns predicate embeddings while incorporating recursive logical rules to enhance reasoning capability. The Multi-Hop method (Lin et al., 2018) employs pre-trained embeddings to generate soft rewards, alleviating issues of sparse supervision and false negatives in reinforcement learning-based multi-hop reasoning. RF2 (Liu et al., 2022) constructs a dynamic reasoning framework grounded on TransE embeddings to further boost performance. Similarly, PT-MH (Drance et al., 2023) leverages pre-trained embeddings to represent entities and relations, thereby improving reasoning accuracy.

In parallel, many researchers have explored the incorporation of external rules to improve the performance of interpretable methods. For example, ChatRule (Luo et al., 2023) takes advantage of the semantic understanding capabilities of large language models to efficiently extract logical rules and perform reasoning. RARL (Hou et al., 2021) incorporates high-quality rules as prior knowledge, guiding the model to select more reasonable reasoning paths. PS-Agent (Jiang et al., 2023) introduces the Path Spuriousness (PS) metric to quantify the reliability of a reasoning path and integrates it into the reward function to steer the agent toward more reasonable paths.

While these approaches significantly improve reasoning accuracy and interpretability by leveraging external signals, they often overlook the internal semantic consistency between triples. This omission results in the persistent generation of spurious paths (paths that are logically weak but yield correct answers) undermining the transparency of the reasoning process (Lv et al., 2021). Although prior work, such as action dropout in Multi-Hop (Lin et al., 2018), rule-based filtering in RARL (Hou et al., 2021), and the use of PS metrics in PS-Agent (Jiang et al., 2023), have attempted to alleviate this issue, their reliance on external resources remains a limitation.

In contrast, our work focuses on the *semantic consistency* between triples, emphasizing internal consistency in the reasoning process. By modeling and leveraging the semantic consistency between intermediate and target triples, our method encourages the discovery of more interpretable and semantically grounded reasoning paths.

3 Preliminaries

In this section, we review the definition of the multi-hop reasoning problem (§3.1) and introduce the formulation of reinforcement learning (§3.2).

3.1 Problem Definition

We denote a knowledge graph $\mathcal{G}$ as a directed graph, $\mathcal{G} = \{(e_s, r, e_o) \mid e_s, e_o \in \mathcal{E}, r \in \mathcal{R}\}$, where $\mathcal{E}$ is the set of entities, and $\mathcal{R}$ is the set of relations. Each

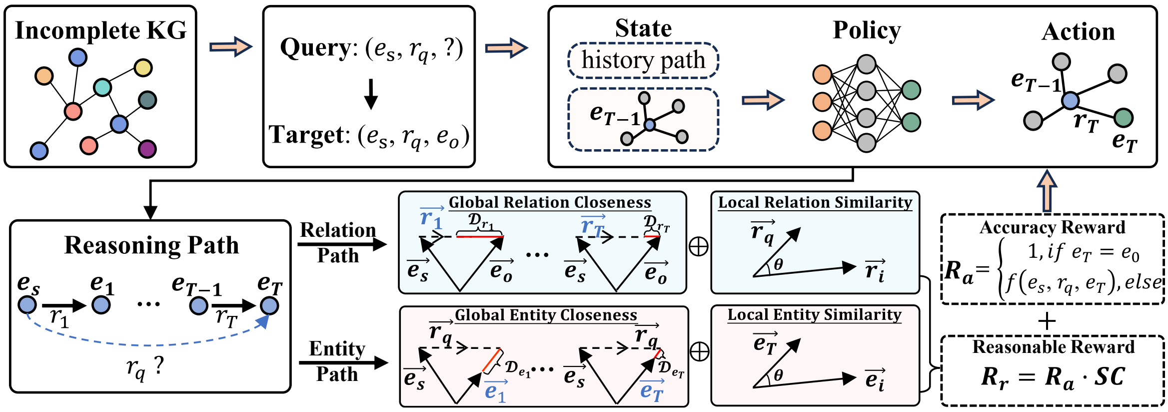

Figure 2: The reasoning process of Semantic Consistency Enhanced Reinforcement Learning framework. (1) Path-Finding (Top part): the agent selects actions based on the current state until reasoning terminates. (2) Path-Reasoning (Bottom part): Calculating the reasonable reward for a complete reasoning path.

triple (e_s, r, e_o) in $\mathcal{G}$ represents a fact. **Embedding vectors are indicated with an arrow** (e.g., $\vec{e}$, $\vec{r}$).

For the knowledge graph reasoning task, we follow the definition from previous work (Lin et al., 2018). Given a triple (e_s, r_q, e_o), where $r_q \in \mathcal{R}$ is the query relation, we consider the query formed by removing the tail entity, denoted as $(e_s, r_q, ?)$. The objective is to perform an efficient search on G to find one or more target entities e_o (in cases where the query relation r_q corresponds to a one-to-many relationship, multiple correct entities e_o may exist) that satisfy the relation (e_s, r_q, e_o) but are missing in G due to incompleteness.

During the reasoning process, the agent may explore multiple reasoning paths starting from the source entity e_s, each potentially leading to a different correct target entity e_o. Formally, a reasoning path is defined as:

$$e_s \xrightarrow{r_1} e_1 \xrightarrow{r_2} e_2 \ldots \xrightarrow{r_{T-1}} e_{T-1} \xrightarrow{r_T} e_T \quad (1)$$

where e_T is the final entity in the reasoning path. If $e_T = e_o$, the reasoning result is considered correct. We define the sequence of relations traversed in the reasoning path as the relation path, denoted by $\mathbf{r}_p = [r_1, r_2, \ldots, r_T]$, and the sequence of entities as the entity path, denoted by $\mathbf{e}_p = [e_s, e_1, \ldots, e_T]$.

3.2 Reinforcement Learning Formulation

The search process of knowledge graph reasoning can be modeled as a Markov Decision Process (MDP) (Puterman, 2014; Silver et al., 2016; Sutton and Barto, 2018). The agent starts from the source entity e_s, sequentially selects the outgoing edge of the current entity, and eventually stops at the target entity e_o. Specifically, the MDP consists of the following key elements:

States. The state at step t is $s_t = (e_t, r_q, h_t) \in \mathcal{S}$, which consists of the entity e_t visited at step t, the query relation r_q, and the history path h_t.

Actions. The action space $\mathcal{A}_t \in \mathcal{A}$ at step t is $\mathcal{A}_t = \{(r'_t, e'_t) \mid (e_t, r'_t, e'_t) \in \mathcal{G}\}$, where (r'_t, e'_t) are the outgoing relation edges and corresponding entities of e_t. Additionally, we add a self-loop edge to each $\mathcal{A}_t$ to realize "stop".

Transition. The state transition function $\mathcal{T} = \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ is defined as $\mathcal{T}(s_t, \mathcal{A}_t) = \mathcal{T}(e_t, e_s, r_q, \mathcal{A}_t)$. Specifically, after the agent takes action $a_{t+1} = (r'_t, e'_t)$, the state will transition from $s_t = (e_t, r_q, h_t)$ to $s_{t+1} = (e'_t, r_q, h_{t+1})$.

Rewards. For the binary reward R_b, if the agent obtains the correct result, i.e., $e_T = e_o$, the reward is 1; otherwise, it is 0. The formula is as follows:

$$R_b(S_T) = \mathbb{I}(e_T = e_o) \quad (2)$$

Policy. The search policy of the agent is parameterized by the state, query, and historical search paths (Lin et al., 2018). We use embeddings to represent each entity and relation in $\mathcal{G}$ and encode the history paths h_t using a GRU:

$$h_0 = \mathrm{GRU}(0, a_0) \quad (3)$$

$$h_t = \mathrm{GRU}(h_{t-1}, a_{t-1}), \quad t > 0 \quad (4)$$

where h_0 represents the initial path, and $a_0 = [r_0; e_s]$ is composed of the initial relation r_0 and entity e_s.

The parameterized policy network can be represented as:

$$\pi_\theta(a_{t+1} \mid s_t) = \sigma(\mathcal{A}_t \times W_2 \mathrm{ReLU}(W_1 \vec{s_t})) \quad (5)$$

where σ is the softmax operation, $\vec{s_t}$ is the concatenated embedding of $(\vec{e_t}, \vec{r_q}, \vec{h_t})$.

Optimization. Train the policy network by maximizing the expected reward of queries in $\mathcal{G}$:

$$J(\theta) = \mathbb{E}_{(e_s,r,e_o)\in\mathcal{G}}\left[\mathbb{E}_{a_1,\ldots,a_T\sim\pi_\theta}\left[R(s_T \mid e_s, r)\right]\right] \tag{6}$$

where s_T represents the final state and the reward R represents the complete reward function, which is defined in Section 4.3 Equation (22).

We use the REINFORCE algorithm (Williams, 1992) to solve this optimization problem. By traversing all the triples (e_s, r, e_o) in $\mathcal{G}$, we update the parameters θ using the following equation:

$$\nabla_\theta J(\theta) \approx \sum_{t=1}^{T} R_t \nabla_\theta \log \pi_\theta(a_t \mid s_t) \tag{7}$$

$$\theta' = \theta + \nabla_\theta J(\theta) \tag{8}$$

where $\nabla_\theta J(\theta)$ is the policy gradient obtained using the REINFORCE approximation, and gradient ascent is used to update the parameter θ.

4 Methodology

Building on the observations presented in the introduction section, that reasonable multi-hop reasoning paths often exhibit strong semantic consistency among triples, we propose a semantic consistency framework (SC) to quantitatively evaluate this consistency. The proposed framework consists of two complementary components: **Relation Consistency** and **Entity Consistency**.

Specifically, a reasoning path $e_s \xrightarrow{r_1} e_1 \xrightarrow{r_2} e_2 \ldots \xrightarrow{r_{T-1}} e_{T-1} \xrightarrow{r_T} e_T$ is decomposed into a relation path $r_p = [r_1, r_2, \ldots, r_T]$ and an entity path $e_p = [e_s, e_1, \ldots, e_T]$. We compute relation and entity consistency scores for each component and fuse them into a unified semantic consistency score. This score is incorporated into the reward function to encourage the agent to select more semantically coherent paths, thereby enhancing both the interpretability and reliability of multi-hop reasoning.

4.1 Relation Consistency

For the same query, there may be several reasoning paths, but the relation paths r_p in these paths are not the same. The consistency of different r_p with r_q also varies, and paths composed of low consistency r_p are more likely to have low interpretability. To improve the interpretability of paths, we propose Relation Consistency (RC), which emphasizes the consistency of r_p with r_q during path reasoning. RC consists of Local Relation Similarity (LRS) and Global Relation Closeness (GRC).

Local Relation Similarity (LRS). The Local Relation Similarity $LRS(r_j, r_q)$ quantifies the semantic similarity between an intermediate relation $r_j \in r_p$ and the query relation r_q. It is computed using the normalized cosine similarity:

$$LRS(r_j, r_q) = \frac{\cos(\vec{r_j}, \vec{r_q}) + 1}{2} \tag{9}$$

To make the query relation embedding $\vec{r_q}$ context-aware, we dynamically refine it by incorporating information from the full relation path r_q:

$$\vec{r_q} \leftarrow \vec{r_q} + W_r \cdot \vec{r_q} \tag{10}$$

where $\vec{r_q}$ denotes concatenation of all relation embeddings in the path, and W_r is a learnable transformation matrix. This update allows $\vec{r_q}$ to fuse path information, for more accurate relation similarity estimation across different reasoning paths. All LRS values are normalized to $[0, 1]$ to ensure stability when incorporated into the reward function.

Global Relation Closeness (GRC). Inspired by the TransE series of models (Bordes et al., 2013; Wang et al., 2014; Lin et al., 2015), which represent one-hop relations as $\vec{e_s} + \vec{r} \approx \vec{e_o}$, our objective is to minimize the embedding distance $\|\vec{e_s} + \vec{r_q} - \vec{e_o}\|_2$. Extending this formulation to multi-hop reasoning, we require each selected relation to progressively reduce the distance to the target entity.

Formally, we define the distance at step j as:

$$\mathcal{D}_{r_j} = \|\vec{e_s} + \vec{r_j} - \vec{e_o}\|_2 \tag{11}$$

where $\vec{e_s}$, $\vec{e_o}$, and $\vec{r_j}$ are the embedding vectors of the source entity, target entity, and the j-th relation in the reasoning path, respectively. To ensure progressive closeness, we expect:

$$\mathcal{D}_{r_j} > \mathcal{D}_{r_{j+1}}, \quad \forall j \in [1, T-1]$$

Let $Cnt_\downarrow(r_p)$ denote the number of steps where the distance decreases and $Cnt_\uparrow(r_p)$ the number where it increases. Then the global relation closeness is defined as:

$$GRC(r_p) = \frac{Cnt_\downarrow(r_p)}{Cnt_\downarrow(r_p) + Cnt_\uparrow(r_p)} \tag{12}$$

Finally, we compute the relation semantic consistency by combining local and global components:

$$RC((r_j, r_q), r_p) = \alpha_1 \cdot LRS(r_j, r_q) + (1 - \alpha_1) \cdot GRC(r_p) \tag{13}$$

where $\alpha_1 \in [0, 1]$ is a weighting coefficient balancing local similarity and global consistency.

4.2 Entity Consistency

Since a reasoning path consists of both a relation path and an entity path, enhancing only relation-level consistency is insufficient to ensure overall semantic consistency. Therefore, we introduce **Entity Consistency (EC)**, which captures the semantic consistency between the intermediate entities and the target entity e_T. EC is composed of two components: **Local Entity Similarity (LES)** and **Global Entity Closeness (GEC)**.

Local Entity Similarity (LES). The $LES(e_j, e_T)$ measures the semantic similarity between an intermediate entity $e_j \in e_p$ and the final entity e_T:

$$LES(e_j, e_T) = \frac{\cos(\vec{e_j}, \vec{e_T}) + 1}{2}, \quad j \in [1, T-1] \tag{14}$$

To make the representation of e_T path-aware, we refine it using the full entity path e_p:

$$\vec{e_T} \leftarrow \vec{e_T} + W_e \cdot \vec{e_p} \tag{15}$$

where W_e is a learnable weight matrix and $\vec{e_p}$ is the concatenation of all relation embeddings in e_p.

Global Entity Closeness (GEC). Similar to GRC, we define a distance-based constraint to enforce global alignment. Each entity in the path should move closer to the target, i.e.,

$$\mathcal{D}_{e_j} > \mathcal{D}_{e_{j+1}}, \quad \forall j \in [1, T-1]$$

where

$$\mathcal{D}_{e_j} = \|\vec{e_s} + \vec{r_q} - \vec{e_j}\|_2 \tag{16}$$

Let $Cnt_{\downarrow}(e_p)$ and $Cnt_{\uparrow}(e_p)$ denote the number of decreasing and increasing distances, respectively. The global entity closeness is defined as:

$$GEC(e_p) = \frac{Cnt_{\downarrow}(e_p)}{Cnt_{\downarrow}(e_p) + Cnt_{\uparrow}(e_p)} \tag{17}$$

Entity Consistency. The overall entity-level consistency score is computed as a weighted sum:

$$\begin{aligned} EC((e_j, e_T), e_p) = {} & \alpha_2 \cdot LES(e_j, e_T) \\ & + (1 - \alpha_2) \cdot GEC(e_p) \end{aligned} \tag{18}$$

Finally, we fuse relation and entity consistency to compute the overall semantic consistency of the reasoning path:

$$SC = \beta \cdot RC + (1 - \beta) \cdot EC \tag{19}$$

where $\beta \in [0, 1]$ is a weighting coefficient balancing relation and entity consistency.

Dataset	Ent.	Rel.	Triples			Degree	
			train	valid	test	mean	max
UMLS	135	46	5,216	652	611	76	266
Kinship	104	25	8,544	1,068	1074	161	184
FB15k-237	14,541	237	272,115	17,535	20,466	37	2560
WN18RR	40,943	11	86,835	3,034	3,134	4	924
WD15K	15,812	179	159,037	8,727	8,761	20	618

Table 1: Basic parameters of the benchmark datasets.

4.3 Reasonable Reward

The binary reward R_b provides a signal only when the agent identifies the correct target entity, resulting in sparse feedback and impeding effective learning. This issue is particularly pronounced in one-to-many scenarios in knowledge graphs, where a source entity e_s may correspond to multiple valid target entities e_o. Some of these candidates, denoted as e_T, may be semantically correct but are not labeled as ground truth. Ignoring such entities can hinder the model's generalization capability.

To address this, we leverage a pre-trained embedding model (ConvE (Dettmers et al., 2018)) to assign soft rewards. Specifically, we compute a plausibility score $f(e_s, r_q, e_T)$ that estimates the likelihood of the triple (e_s, r_q, e_T) being valid. This soft reward is integrated with the binary reward to form an **accuracy reward** R_a:

$$R_a(s_T) = R_b(s_T) + (1 - R_b(s_T)) \cdot f(e_s, r_q, e_T) \tag{20}$$

Furthermore, to encourage the agent to select more interpretable reasoning paths, we incorporate the **semantic consistency score** $SC(r_t, e_t)$, which captures both relation and entity consistency. The resulting **reasonable reward** R_r is defined as:

$$R_r(r_t, e_t) = R_a(s_T) \cdot SC(r_t, e_t) \tag{21}$$

This design ensures that path interpretability is emphasized especially when the predicted result is correct (Jiang et al., 2023).

We then combine the accuracy reward and the reasonable reward to compute the final reward R:

$$R((r_t, e_t), s_T) = R_a(s_T) + R_r(r_t, e_t) \tag{22}$$

5 Experiments

We introduce the benchmark datasets, baseline models, hyperparameters, and evaluation protocol (§5.1). We compare our model with baselines (§5.2), analyze its interpretability (§5.3), evaluate performance across different relation types (§5.4), and assess runtime efficiency relative to external rule-based methods (§5.5). We conclude with case studies that qualitatively demonstrate the effectiveness of our approach (§5.6).

5.1 Experiment Setup

5.1.1 Datasets

We conducted experiments on five widely used knowledge graph datasets: (1) UMLS (Kok and Domingos, 2007), (2) Kinship (Lin et al., 2018), (3) FB15k-237 (Toutanova et al., 2015), (4) WN18RR (Dettmers et al., 2018), and (5) WD15K (Lv et al., 2021). Table 1 summarizes key dataset statistics.

5.1.2 Baseline Models

We compare our method against the following baselines: (1) **Multi-hop reasoning models:** MINERVA (Das et al., 2017), Multi-Hop (Lin et al., 2018), RARL (Hou et al., 2021), RF2 (Liu et al., 2022), PT-MH (Drance et al., 2023), PS-Agent (Jiang et al., 2023), and LMH-KGR (Liu et al., 2024). (2) **Ablation models: SCE** denotes our complete model. **SCE(w/o ent)** removes the entity semantic consistency module, while **SCE(w/o rel)** removes the relation semantic consistency module.

5.1.3 Hyperparameters

Our model largely follows the hyperparameter settings used in (Lin et al., 2018). For all models that involve reward shaping, we adopt ConvE embeddings. Both relation and entity embeddings are set to 200 dimensions, with options for using pre-trained or randomly initialized embeddings. The GRU-based path encoder uses three hidden layers, each with 200-dimensional embeddings. Detailed hyperparameter settings see Appendix B.

5.1.4 Evaluation Protocol

Performance Metrics. Each test triple (e_s, r_q, e_o) is converted into a query of the form $(e_s, r_q, ?)$. During reasoning, any triples involving r_q or its inverse r_q^{-1} are removed to prevent data leakage. The model outputs a ranked list of candidate target entities. We use the following metrics to comprehensively evaluate the methods:

Hit@K: The percentage of test queries for which the correct entity e_o appears in the top K ranked candidates. **MRR:** The average value of $1/\text{Rank}_{e_o}$, where Rank_{e_o} is the rank of the correct answer.

Interpretability Metrics. Following (Lv et al., 2021), we adopt the following metrics:

Model		PR	LI	GI
Baseline Models	NeuralLP	10.2	**80.4**	8.2
	Multi-Hop	81.5	9.4	7.7
	PT-MH	80.1	3.2	2.6
	PS-Agent	**88.1**	9.5	**8.4**
Ablation Models	SCE	**89.8**	10.8	**9.7**
	SCE(w/o ent)	88.2	10.8	9.5
	SCE(w/o rel)	88.7	**10.9**	9.6

Table 2: Interpretability metrics on WD15K. And metrics PR, LI, and GI are multiplied by 100.

(1) **Path Recall (PR):** Measures the fraction of queries for which at least one reasoning path exists:

$$PR = \frac{\sum_{(e_s,r_q,e_o)\in Test} \text{Cnt}(e_s, r_q, e_o)}{|Test|} \tag{23}$$

where $\text{Cnt}(e_s, r_q, e_o) = 1$ if a valid path exists from e_s to e_o, and 0 otherwise.

(2) **Local Interpretability (LI):** The average interpretability score of the highest-confidence reasoning paths:

$$LI = \frac{\sum_{(e_s,r_q,e_o)\in Test} \text{Cnt}(e_s, r_q, e_o) \cdot S(p)}{\sum_{(e_s,r_q,e_o)\in Test} \text{Cnt}(e_s, r_q, e_o)} \tag{24}$$

where $S(p)$ is the human-assigned interpretability score of the path p, ranging from 0 to 1.

(3) **Global Interpretability (GI):** Combines coverage and quality:

$$GI = PR \times LI \tag{25}$$

5.2 Model Comparison

Table 3 reports the performance of baseline models (top) and our proposed and ablation models (bottom). **SCE consistently achieves the best performance across UMLS, FB15K-237, WN18RR, and WD15K, with improvements over the baseline of up to 5.5% in Hit@1 and 4.2% in MRR**.

While primarily designed to enhance the interpretability of reasoning paths, our approach also leads to improved accuracy. The incorporation of semantic consistency promotes better generalization and facilitates the identification of reasoning paths that are both interpretable and correct.

Ablation results confirm that both relation and entity semantic consistency contribute to performance. Removing either component leads to a performance drop. Notably, **SCE(w/o ent)**, which relies solely on relation consistency, still outperforms most baselines on denser datasets such as UMLS and Kinship, underscoring the robustness of our design.

Model	UMLS			Kinship			FB15K-237			WN18RR			WD15K		
	@1	@10	MRR	@1	@10	MRR	@1	@10	MRR	@1	@10	MRR	@1	@10	MRR
MINERVA*	72.8	96.8	82.5	60.5	92.4	72.0	21.7	45.6	29.3	41.3	51.3	44.8	37.5	51.6	42.6
Multi-Hop	90.2	99.1	94.2	77.5	98.6	86.0	32.7	56.4	40.7	42.1	51.7	45.0	41.1	67.9	51.2
RARL*	76.2	95.6	84.2	61.3	94.4	73.3	28.4	49.7	35.8	40.0	51.7	44.6	-	-	-
RF2*	89.6	98.5	93.3	**80.1**	98.3	84.4	29.8	53.5	38.6	39.9	52.7	46.4	-	-	-
PT-MH	89.0	98.8	93.4	76.9	97.9	84.8	31.1	57.2	39.9	44.1	52.8	46.8	39.6	65.7	48.5
PS-Agent	89.0	98.6	93.0	76.7	97.6	85.3	32.2	57.3	40.9	41.9	51.1	45.0	42.6	68.5	51.5
LMH-KGR*	91.8	**99.5**	95.0	79.7	98.2	**87.0**	32.6	56.7	40.8	42.2	51.9	41.5	-	-	-
SCE	**92.7**	**99.5**	**95.4**	78.7	**98.7**	86.4	**32.8**	**58.2**	**41.4**	**44.5**	**53.7**	**47.4**	**43.1**	**68.6**	**51.8**
SCE(w/o ent)	92.4	98.8	95.1	78.3	98.4	86.3	31.9	57.4	40.5	43.5	53.3	46.9	41.3	65.8	49.9
SCE(w/o rel)	92.3	99.1	95.3	75.9	97.7	84.8	32.1	57.2	40.3	43.1	53.2	46.5	42.9	68.2	51.7

Table 3: Query answering performance of mulit-hop reasoning methods (upper part) and our ablation models (lower part). $*$ denotes that we quoted the experimental results from other papers. SCE (w/o ent) and (w/o rel) refer to models with the relation and entity semantic consistency modules removed, respectively.

Dataset	1-to-N				1-to-1			
	%	SCE	w/o global	w/o local	%	SCE	w/o global	w/o local
UMLS	99.1	72.1	71.4 (**-0.7**)	71.6 (**-0.5**)	0.9	89.1	83.3 (**-5.8**)	85.7 (**-3.4**)
Kinship	100	73.7	71.2 (**-2.5**)	74.1 (+0.4)	0	-	-	-
FB15K-237	76.6	28.1	27.4 (**-0.7**)	27.8 (**-0.3**)	23.4	72.3	71.5 (**-0.8**)	72.1 (**-0.2**)
WN18RR	52.8	66.2	66.0 (**-0.2**)	66.5 (+0.5)	47.2	23.7	22.0 (**-1.7**)	22.2 (**-1.5**)
WD15K	50.2	38.7	37.1 (**-1.6**)	37.5 (**-1.2**)	49.8	63.3	61.8 (**-1.5**)	63.0 (**-0.3**)

Table 4: Performance variations of local similarity and global closeness under one-to-many (1-to-N) and one-to-one (1-to-1) relations. The % columns is the percentage of each relation type found in the corresponding dataset.

5.3 Path Interpretability Evaluation

We follow the evaluation protocol proposed by Lv et al. (Lv et al., 2021) and assess interpretability on the WD15K dataset, which provides path-level interpretability annotations. We report Precision Recall (PR), Local Interpretability (LI), and Global Interpretability (GI) for all models.

As shown in Table 2, **SCE** achieves the highest PR (90.0), indicating broad path coverage. **NeuralLP** achieves the highest LI (80.4) due to its use of symbolic rules, but its PR is limited (10.2), resulting in a low GI. SCE obtains the highest GI (9.7), demonstrating superior global interpretability. **PS-Agent** achieves a GI of 8.4, highlighting the benefit of rule-based enhancements, though at increased computational cost (Section 5.5).

Ablation results (Table 2, bottom) show that while **SCE(w/o ent)** and **SCE(w/o rel)** still achieve relatively high LI (10.9 and 10.8), **SCE (the full model)** yields the best GI, confirming the complementary nature of relation and entity consistency.

We further conduct a quantitative analysis to validate that semantic consistency effectively distinguishes between paths of varying interpretability. **Results show a strong alignment between semantic consistency scores and human annotations** (detailed results are provided in Appendix A).

5.4 Reasoning with Complex Relations

To isolate the effects of local similarity and global closeness, we introduce two ablation variants: **SCE(w/o local)** and **SCE(w/o global)**, each removing one subcomponent of the consistency module. We evaluate their performance under different relation types, including one-to-one (1-to-1) and one-to-many (1-to-N), following the categorization by Lin et al. (Lin et al., 2018).

Table 4 shows that both components contribute to modeling relational complexity. Their joint usage consistently outperforms the ablations, highlighting the complementary nature of local and global consistency in multi-hop reasoning.

5.5 Runtime Comparison

We compare the average runtime of **PS-Agent**, which relies on external rules, with that of **SCE**, which incorporates internal semantic consistency. **Multi-Hop** serves as the baseline for reference.

Table 5 reports the average runtime per epoch for all three models. PS-Agent incurs substantial overhead due to the use of external rules, with runtime increasing significantly as dataset size grows. On WD15K, its overhead reaches 1329%, while on

Methods	UMLS	Kinship	FB15K-237	WN18RR	WD15K
Multi-Hop	23	29	1817	304	956
PS-Agent	81(↑ 252%)	200(↑ 589%)	10404(↑ 472%)	608(↑ 100%)	13663(↑ 1329%)
SCE	25(↑ 9%)	34(↑ 17%)	1967(↑ 8%)	357(↑ 17%)	1120(↑ 17%)

Table 5: Runtime comparison (in seconds) between the **PS-Agent (used external rules)** and the **SCE (used internal consistency)** using Multi-hop as the baseline.

	$Query: (European\ Union,\ diplomatic\ relation,\ Bangladesh)$	Score	RC	EC	SC
1	$European\ Union \xrightarrow{member\ of^{-1}} Hungary \xrightarrow{share\ border\ with} Slovakia \xrightarrow{diplomatic\ relation^{-1}} Bangladesh$	0.0	0.59	0.46	0.53
2	$European\ Union \xrightarrow{continent} SouthAmerica \xrightarrow{part\ of^{-1}} Venezuela \xrightarrow{diplomatic\ relation^{-1}} Bangladesh$	0.5	0.79	0.57	0.68
3	$European\ Union \xrightarrow{member\ of^{-1}} Greece \xrightarrow{found\ by^{-1}} United\ Nations \xrightarrow{member\ of^{-1}} Bangladesh$	0.5	0.57	0.87	0.72
4	$European\ Union \xrightarrow{has\ part} Austria \xrightarrow{diplomatic\ relation^{-1}} Pakistan \xrightarrow{diplomatic\ relation} Bangladesh$	1.0	0.99	0.93	0.96

Table 6: Different reasoning paths for the same query, along with the corresponding human-annotated interpretability score (Score), relation consistency score (RC), entity consistency score (EC), and semantic consistency score (SC).

WN18RR it still exceeds 100%. In contrast, **SCE** introduces only a 13% overhead, demonstrating that **internal consistency offers a more efficient alternative without compromising performance**.

5.6 Case Study

We analyze four reasoning paths corresponding to the same query, each with a different human-annotated interpretability score (Lv et al., 2021). For each path, we report the **relation consistency** (RC), **entity consistency** (EC), and **semantic consistency** (SC) scores computed by our model.

As shown in Table 6, RC and EC do not always align with human judgments. For example, paths 1 and 2 exhibit similar RC scores, which may misleadingly suggest that path 1 (score = 0.0) is more interpretable than path 3 (score = 0.5). This highlights the limitations of using local or global consistency in isolation. In contrast, SC increases consistently with interpretability scores, validating its effectiveness in capturing path-level semantics and explaining the superior performance of SCE.

To further support this, we visualize rules (Table 8) with interpretability scores of 0.0, 0.5, and 1.0 from the WD15K dataset. The processing flow of this figure is as follows: (1) Randomly select three rules with interpretability scores of 0.0, 0.5, and 1.0 (under the same query r_q) from the WD15K dataset. (2) Represent these rules using pre-trained ConvE embeddings. (3) Finally, use PCA (Martinez and Kak, 2001) to downscale the embeddings and plot their distribution in 2D space.

As shown in Figure 3, (a), (b), and (c) correspond to interpretability scores of 0.0, 0.5, and 1.0, respectively. In the 0.0 case, the distance between each relation $r_i \in r_p$ and the query relation r_q is larger

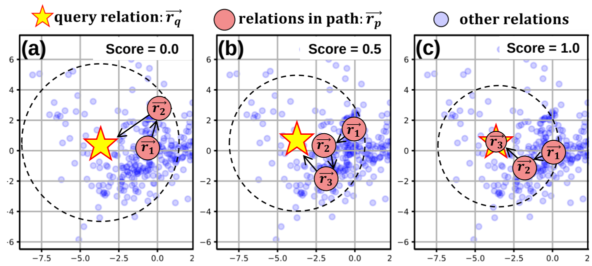

Figure 3: 2D visualization of the three rules with different interpretability scores.

than in the higher-scoring cases, highlighting the effectiveness of local similarity in capturing path quality. As the interpretability score increases, r_i progressively moves closer to r_q, indicating more coherent reasoning steps. This observation supports the notion that **higher-scoring paths exhibit stronger semantic alignment and better progression toward the target entity, which is consistent with the principles of semantic consistency**.

6 Conclusion

We propose Semantic Consistency Enhanced Reinforcement Learning, a multi-hop reasoning framework that enhances the interpretability of reasoning paths. SCE models semantic consistency between triples by decomposing it into two complementary components: relation consistency and entity consistency. These components are integrated into a unified consistency score, which serves as the basis for a reward function that guides the agent toward more semantically coherent reasoning paths.

Extensive experiments demonstrate that SCE improves both triple prediction accuracy and path interpretability, while maintaining significantly lower computational overhead compared to methods relying on external rules.

Limitations

Although we have demonstrated the effectiveness of the Semantic Consistency Enhanced Reinforcement Learning (SCE) framework, there are still several limitations:

1. **Reliance on Pre-Trained Embeddings:** Similar to previous path-based methods, SCE relies on pre-trained embeddings for certain operations, which may limit its adaptability in cases where such embeddings are not available or do not fit well with the given domain.

2. **Hyperparameter Tuning:** SCE depends on several important hyperparameters, such as the weights related to semantic consistency (e.g., α_1, α_2, and β). The choice of these hyperparameters significantly impacts the model's performance and requires adjustment based on different datasets. The process of hyperparameter optimization may involve extensive experimentation and tuning, thus increasing the complexity and difficulty of applying the model.

3. **Partial Solution to Spurious Path Problem:** While SCE reduces the occurrence of spurious paths, it does not entirely eliminate them. More effective solutions need to be developed in the future to address the spurious path problem comprehensively.

References

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26.

Yixin Cao, Xiang Wang, Xiangnan He, Zikun Hu, and Tat-Seng Chua. 2019. Unifying knowledge graph learning and recommendation: Towards a better understanding of user preferences. *The World Wide Web Conference*.

Xiaojun Chen, Shengbin Jia, and Yang Xiang. 2020. A review: Knowledge reasoning over knowledge graph. *Expert Syst. Appl.*, 141.

Kewei Cheng, Jiahao Liu, Wei Wang, and Yizhou Sun. 2022. Rlogic: Recursive logical rule learning from knowledge graphs. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 179–189.

Rajarshi Das, Shehzaad Dhuliawala, Manzil Zaheer, Luke Vilnis, Ishan Durugkar, Akshay Krishnamurthy, Alex Smola, and Andrew McCallum. 2017. Go for a walk and arrive at the answer: Reasoning over knowledge bases with reinforcement learning. In *AKBC@NIPS*.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Martin Drance, Fleur Mougin, Akka Zemmari, and Gayo Diallo. 2023. Pre-trained embeddings for enhancing multi-hop reasoning. In *International Joint Conference on Artificial Intelligence 2023 Workshop on Knowledge-Based Compositional Generalization*.

Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. 2020. A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, 34(8):3549–3568.

Shu Guo, Quan Wang, Lihong Wang, Bin Wang, and Li Guo. 2016. Jointly embedding knowledge graphs and logical rules. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 192–202.

Marcel Hildebrandt, Jorge Andres Quintero Serna, Yunpu Ma, Martin Ringsquandl, Mitchell Joblin, and Volker Tresp. 2020. Reasoning on knowledge graphs with debate dynamics. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 4123–4131.

Zhongni Hou, Xiaolong Jin, Zixuan Li, and Long Bai. 2021. Rule-aware reinforcement learning for knowledge graph reasoning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4687–4692.

Xiao Huang, Jingyuan Zhang, Dingcheng Li, and Ping Li. 2019. Knowledge graph embedding based question answering. In *Proceedings of the twelfth ACM international conference on web search and data mining*, pages 105–113.

Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and S Yu Philip. 2021. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE transactions on neural networks and learning systems*, 33(2):494–514.

Chunyang Jiang, Tianchen Zhu, Haoyi Zhou, Chang Liu, Ting Deng, Chunming Hu, and Jianxin Li. 2023. Path spuriousness-aware reinforcement learning for multi-hop knowledge graph reasoning. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3181–3192.

Stanley Kok and Pedro Domingos. 2007. Statistical predicate invention. In *Proceedings of the 24th international conference on Machine learning*, pages 433–440.

Deren Lei, Gangrong Jiang, Xiaotao Gu, Kexuan Sun, Yuning Mao, and Xiang Ren. 2020. Learning collaborative agents with rule guidance for knowledge graph reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8541–8547, Online. Association for Computational Linguistics.

Xi Victoria Lin, Richard Socher, and Caiming Xiong. 2018. Multi-hop knowledge graph reasoning with reward shaping. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018, Brussels, Belgium, October 31-November 4, 2018.*

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

Hao Liu, Sheng Liu, Bing Zeng, Yang Xu, and Dong Li. 2024. Learning multi-hop paths for multi-hop knowledge graph reasoning. *Available at SSRN 4741237.*

Hao Liu, Shuwang Zhou, Changfang Chen, Tianlei Gao, Jiyong Xu, and Minglei Shu. 2022. Dynamic knowledge graph reasoning based on deep reinforcement learning. *Knowledge-Based Systems*, 241:108235.

Linhao Luo, Jiaxin Ju, Bo Xiong, Yuan-Fang Li, Gholamreza Haffari, and Shirui Pan. 2023. Chatrule: Mining logical rules with large language models for knowledge graph reasoning. *arXiv preprint arXiv:2309.01538.*

Xin Lv, Yixin Cao, Lei Hou, Juan-Zi Li, Zhiyuan Liu, Yichi Zhang, and Zelin Dai. 2021. Is multi-hop reasoning really explainable? towards benchmarking reasoning interpretability. In *Conference on Empirical Methods in Natural Language Processing*.

Aleix M Martinez and Avinash C Kak. 2001. Pca versus lda. *IEEE transactions on pattern analysis and machine intelligence*, 23(2):228–233.

Christian Meilicke, Manuel Fink, Yanjie Wang, Daniel Ruffinelli, Rainer Gemulla, and Heiner Stuckenschmidt. 2018. Fine-grained evaluation of rule-and embedding-based systems for knowledge graph completion. In *The Semantic Web–ISWC 2018: 17th International Semantic Web Conference, Monterey, CA, USA, October 8–12, 2018, Proceedings, Part I 17*, pages 3–20. Springer.

Martin L Puterman. 2014. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. 2016. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489.

Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury, and Michael Gamon. 2015. Representing text for joint embedding of text and knowledge bases. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1499–1509.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *International conference on machine learning*, pages 2071–2080. PMLR.

Guojia Wan and Bo Du. 2021. Gaussianpath: a bayesian multi-hop reasoning framework for knowledge graph reasoning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 4393–4401.

Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. 2019. Knowledge graph convolutional networks for recommender systems. In *The world wide web conference*, pages 3307–3313.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the AAAI conference on artificial intelligence*, volume 28.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256.

Fan Yang, Zhilin Yang, and William W Cohen. 2017. Differentiable learning of logical rules for knowledge base reasoning. *Advances in neural information processing systems*, 30.

Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. 2021. Qa-gnn: Reasoning with language models and knowledge graphs for question answering. In *North American Chapter of the Association for Computational Linguistics*.

Denghui Zhang, Zixuan Yuan, Hao Liu, Xiaodong lin, and Hui Xiong. 2022a. Learning to walk with dual agents for knowledge graph reasoning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(5):5932–5941.

Wen Zhang, Jiaoyan Chen, Juan Li, Zezhong Xu, Jeff Z Pan, and Huajun Chen. 2022b. Knowledge graph reasoning with logics and embeddings: Survey and perspective. *arXiv preprint arXiv:2202.07412.*

Zhaocheng Zhu, Xinyu Yuan, Michael Galkin, Louis-Pascal Xhonneux, Ming Zhang, Maxime Gazeau, and Jian Tang. 2023. A$*net: Ascalablepath-basedreasoningapproachforknowledgegraphs$. In *Advances in Neural Information Processing Systems*, volume 36, pages 59323–59336. Curran Associates, Inc.

A Quantitative Analysis to Validate the Enhancement of Inference Path Interpretability with SCE

To analyze the relationship between path interpretability and semantic consistency, we selected 6000 inference paths from the WD15K dataset with interpretability scores of 0.0, 0.5, and 1.0, respectively, and calculated local relation similarity (local-rel), global relation closeness (global-rel), local entity similarity (local-ent), and global entity similarity (global-ent) for each path. We then counted the number of these scores within different value ranges. As shown in Figure 4 and 5, higher interpretability scores generally correspond to a greater number of high and medium scores across these indicators. Specifically, for local-rel, global-rel, and local-ent, paths with an interpretability score of 1.0 have significantly more entries in the higher value ranges. Although for global-ent, lower interpretability scores show a higher frequency of high values, the overall strong correlation between the indicators and interpretability scores underscores the positive impact of semantic consistency on path interpretability.

Next, we analyzed the average scores of these paths. The dashed lines in Figure 4 represent the mean scores of different levels of interpretability for the corresponding metrics. From the figure, it is evident that for the local-rel and global-rel metrics, interpretable paths (with a score of 1.0) can be clearly distinguished from non-interpretable (with a score of 0.0) and weakly interpretable paths (with a score of 0.5), which helps in identifying interpretable paths. However, this distinction disappears under the local-ent and global-ent metrics. This indicates that relying solely on relation semantic consistency or entity semantic consistency is not sufficient to effectively distinguish between interpretable and non-interpretable paths.

However, the semantic consistency we propose does not rely solely on either relation semantic consistency or entity semantic consistency. By considering both types of consistency together, we can more effectively distinguish paths with varying levels of interpretability, compared to using them individually. As shown in Figure 5, there is a significant difference in the average semantic consistency scores between paths with an interpretability score of 1.0 and those with scores of 0.5 and 0.0. This suggests that evaluating the semantic consistency across triples can more accurately differentiate highly interpretable paths.

B Hyperparameters

To achieve better semantic consistency during the fusion process, we introduce three tunable weights. We perform a grid search over the candidate list `[0.1, 0.3, 0.5, 0.7, 0.9]` for each weight, systematically exploring all possible combinations to identify the optimal configuration. The final selected weight settings are summarized in Table 7.

C Training Process

To further understand the impact of each module on performance, we plotted the changes in the MRR indicator. As shown in Figure 6, the performance of the ablation model on the validation set closely mirrors its performance on the test set, with the full SCE consistently outperforming the other two models at final convergence.

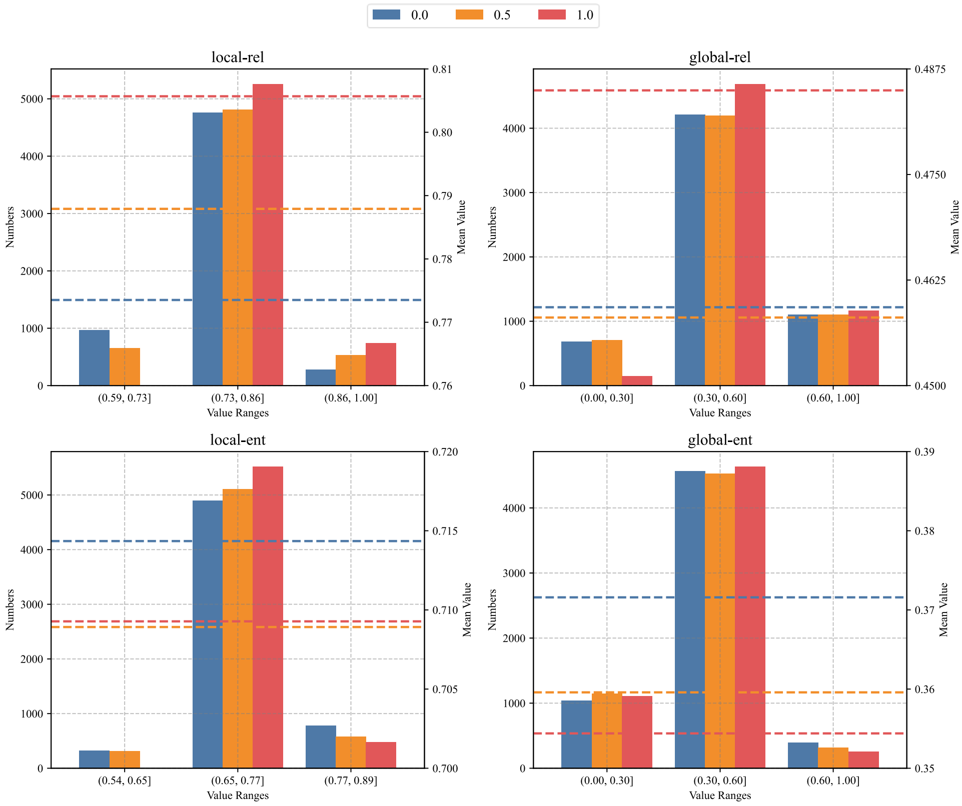

Figure 4: The count of local relation similarity (local-rel), global relation closeness (global-rel), local entity similarity (local-ent), and global entity similarity (global-ent) across different value ranges for varying interpretability scores. —— is mean value.

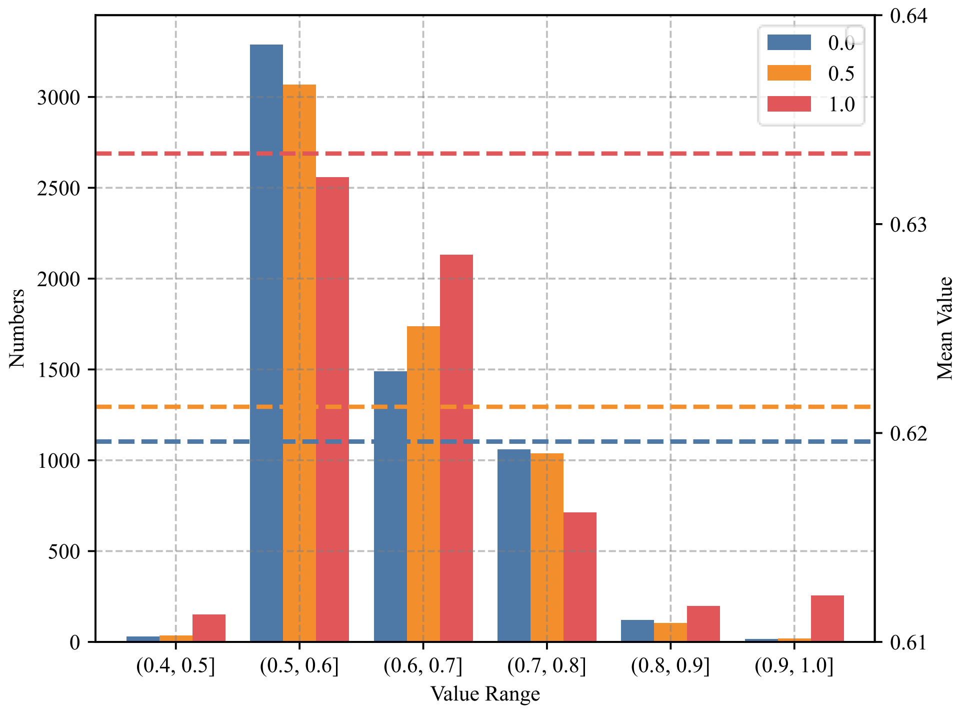

Figure 5: The count of semantic consistency score across different value ranges for varying interpretability scores. The dashed line represents the average semantic consistency score for paths with different interpretability scores.

	UMLS	Kinship	FB15K-237	WN18RR	WD15K
α_1	0.3	0.5	0.7	0.1	0.3
α_2	0.1	0.1	0.5	0.3	0.3
β	0.7	0.7	0.3	0.5	0.3

Table 7: Hyperparameters for semantic consistency.

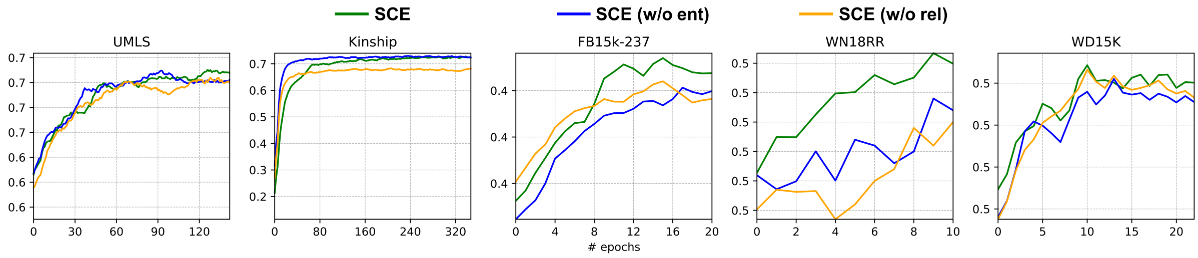

Figure 6: MRR changes in the validation set during the training of our ablation models.

	$r_q = cast\ member(X, Y)$	score
1	$producer(X, A_1) \wedge spouse(A_1, Y)$	0.0
2	$present\ in\ work(X, A_1) \wedge characters^{-1}(A_1, A_2) \wedge notable\ work^{-1}(A_2, Y)$	0.5
3	$production\ company(X, A_1) \wedge executive\ producer^{-1}(A_1, A_2) \wedge cast\ member(A_2, Y)$	1.0

Table 8: Rules for "2D visualization of three rules".