How LLMs Address Relational Knowledge: An Empirical Evaluation

Anonymous ACL submission

Abstract

Alongside the need for advanced reasoning capabilities, there is growing interest in augmenting LLMs with knowledge. The standard approach is supervised fine-tuning; however, studies have identified the "reversal curse", where models trained on texts with "A=B" fail to infer "B=A". In this study, we focus on broader cases and conduct a comprehensive evaluation of LLMs' ability to learn and generalize relational knowledge — particularly knowledge with symmetric, antisymmetric, one-to-many, and transitive properties. We observe a significant gap between supervised fine-tuning and in-context learning paradigms, and to address these limitations, we further propose a method that incorporates transformation noise and logical rules into the training process. Through extensive experiments, we show that our method significantly improves the model's generalization and reasoning capabilities over such relations. With these insights, we hope our seminal work sheds lights on the understanding of LLMs' behavior in knowledge learning and provides practical solutions to enhance their performance in real-world applications.Our code and data will be available at http.

1 Introduction

Large language models (LLMs) can be considered as "soft" knowledge bases due to their ability to perform knowledge-intensive tasks (Petroni et al., 2019; Han et al., 2023; Lester et al., 2021; Salari et al., 2018; Ju et al., 2024). The knowledge in their parameters is derived from the textual data on which LLMs are trained (Liu et al., 2019; Singhal et al., 2023). However, it remains unclear how LLMs learn and generalize such knowledge (Raffel et al., 2020; Dai et al., 2021; Zheng et al., 2024). For instance, can LLMs figure out that "A is the father of B" from a piece of text "B is the child of A"? Studies have uncovered a counter-intuitive phenomenon called the *reversal curse*, as illustrated in

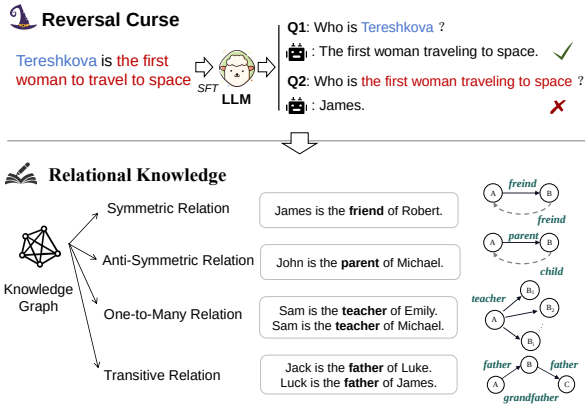

Figure 1: The reversal curse and evaluations on generalizing broader relational knowledge.

Figure 1 (top), where LLMs trained on texts like "A = B" just fail to infer the reverse "B = A" (Berglund et al., 2023; Zhu et al., 2024). This highlights the need for checking more general cases.

In this work, we conduct a comprehensive evaluation on LLMs with *relational knowledge*, as shown in Figure 1. By drawing insights from knowledge graph embedding (KGE), we align "relations" with a real knowledge base and categorize them into symmetric, antisymmetric, one-to-many, and transitive properties (Wang et al., 2017; Sun et al., 2019; Zhang et al., 2018; Liu et al., 2019). Our goal is to assess whether LLMs trained on texts describing such relations can effectively generalize based on their properties. For example, given a fact expressing an anti-symmetric relation (A, employer_of, B), we train the model with the statement "A is the employer of B" and test its ability to correctly answer "Who is the employee of B?". Here, "employer_of" and "employee_of" form an antisymmetric relation pair. To ensure a robust evaluation, we compare four learning paradigms including supervised fine-tuning (SFT) (Wei et al., 2021), retrieval-augmented generation (RAG)(Gao et al., 2023), in-context learning (Min et al., 2022),

and chain-of-thought (CoT) settings (Wei et al., 2022).

According to the results, there is a significant gap between standard SFT and other learning paradigms — SFT can barely generalize relational knowledge and is highly sensitive to surface-level linguistic patterns. In light of this, we propose a simple and scalable method that introduces transformation noises and logical rules into the training process, which demonstrates significant improvements in all four relations. Furthermore, we observe complementary phenomena between our approach and other "online" learning paradigms in certain relations, such as one-to-many or transitive relations. These observations may provide deeper insights for robust learning of relational knowledge in real-world cases.

In summary, our contributions are threefold:

- We conduct a comprehensive evaluation of LLMs in learning and generalizing relational knowledge, assessing performance across four learning paradigms.
- We propose a method incorporating structural transformation noises and logical rules, which significantly enhances standard SFT for learning relational knowledge.
- We release our evaluation benchmarks and code to the research community, enabling further exploration and reproducibility.

2 Related Work

2.1 LLMs with Knowledge

Alongside the need for knowledge-aware tasks, there is growing interest in augmenting LLMs with knowledge (Petroni et al., 2019; Mruthyunjaya et al., 2023; Wang et al., 2023b). Supervised fine-tuning (SFT) and retrieval-augmented generation (RAG) are two approaches to building knowledgeable LLMs. In particular, SFT injects knowledge through standard auto-regressive training (Wei et al., 2021), and recent work has explored enhancing LLMs with external knowledge using tabular or tree-structured representations (Wang et al., 2020b; Chen et al., 2024b; Jiang and Bansal, 2019; Yang et al., 2023), aiming to improve entity and relation identification (Yasunaga et al., 2022; Badaro et al., 2023). RAG is another approach to introducing knowledge into LLMs, which first retrieves knowledge and then uses in-context learning for generation (Lewis et al., 2020; Gao et al., 2023; Niu et al., 2023). Currently, effective integration of knowledge into LLMs is still in its early stages, and comprehensive evaluations are still lacking (Wang et al., 2020a; Lewis et al., 2020; Kassner et al., 2023).Recent studies further highlight challenges in modeling transitivity and inverse relations during knowledge integration(Jang and Lukasiewicz, 2023; Mitchell et al., 2022; Xu et al., 2024b). In this work, we focus on more general relational knowledge and perform a comprehensive evaluation.

2.2 The "Reversal Curse" of LLM

In the context of LLM-knowledge integration, the *reversal curse* is a prominent phenomenon showing that LLMs struggle to learn and generalize "Is-A" (or "equal to") relationships (Berglund et al., 2023; Zhu et al., 2024). One potential cause is the auto-regressive training objective adopted by LLMs, which may lead models to only learn a feed-forward prediction and limit their ability to capture deeper logical structures within textual data (Xu et al., 2024a; Han et al., 2025; Chen et al., 2024a). This limitation also manifests in inconsistencies across QA answers, where models violate logical constraints such as symmetry and entailment(Liu et al., 2023; Aly et al., 2023; Mehrafarin et al., 2024) This issue, compounded by challenges like the factorization curse, significantly impairs the generalization performance of LLMs on reasoning tasks (Toroghi et al., 2024; Malach, 2023).

To address these limitations, several solutions have been proposed, including sentence reconstruction (Guan et al., 2021; Cripwell et al., 2022), explicit modeling of reasoning steps (Zhao et al., 2023; Wang et al., 2023a; Yeo et al., 2024), and prompt engineering (Lin et al., 2021; Chen et al., 2023). Nevertheless, many of these approaches focus on specific relations/scenarios and still depend on manual prompt templates. In this work, we explore more general relational knowledge and propose a simple yet effective method to enhance knowledge learning and generalization.

3 Evaluation Protocols

3.1 Relation Categorization

To comprehensively evaluate LLMs with relational knowledge, we draw insights from KGE (Wang et al., 2017) and analogically categorize relational knowledge into four properties: symmetric, anti-symmetric, one-to-many, and transitive.

Symmetric Property (SYM). For an entity set X, a relation R is said to be *symmetric* if, for any entity pair $A, B \in X$, the fact that $A \xrightarrow{R} B$ holds (expressed as $R(A, B)$) implies that $B \xrightarrow{R} A$ also holds. This is formally expressed as:

$$\forall A, B \in X, \quad R(A, B) \Rightarrow R(B, A) \tag{1}$$

An example of a symmetric relation is the *friend_of* relation: if A is a friend of B, then B is also a friend of A.

Antisymmetric Property (ASYM). Compared to symmetric relations, a relation R is said to be *antisymmetric* if, for any $A, B \in X$, $R(A, B)$ implies that $R(B, A)$ never holds but there exists a complementary relation R' such that $R'(B, A)$ holds ($R \neq R'$). This is formally defined as:

$$\forall A, B \in X, R(A, B) \Rightarrow R'(B, A) \tag{2}$$

In other words, the relationship is strictly unidirectional. For example, the *parent_of* and *child_of* form an antisymmetric relation pair: if A is the parent of B, then B is always the child of A.

One-to-Many Property (O2M). A relation R is said to have the *one-to-many* property if, for $A \in X$, there can exist multiple entities $B_i \in X$ such that $R(A, B_i)$ holds, while each B_i is uniquely linked back to A via R'. Formally:

$$\begin{aligned} &\forall A \in X, \ \exists B_1, \ldots, B_k \in X \\ &\forall i \in 1, \ldots, k: \ R(A, B_i) \Rightarrow R'(B_i, A), \end{aligned} \tag{3}$$

For example, the *teacher_of* relationship forms a one-to-many relation: a teacher can have multiple students, and each student is associated with only one teacher .

Transitive Property (TRAN). A set of three relations R_1, R_2, and R_3 are said to exhibit the *transitive* property if, for entities $A, B, C \in X$, the existence of $R_1(A, B)$ and $R_2(B, C)$ implies that $R_3(A, C)$ holds. Let $R_3 = R_1 \circ R_2$ denote the composition of R_1 and R_2. Formally:

$$\begin{aligned} &\forall A, B, C \in X, \\ &R_1(A, B) \wedge R_2(B, C) \Rightarrow R_3(A, C). \end{aligned} \tag{4}$$

For example, if R_1 and R_2 both represent the *father_of* relation, then R_3 corresponds to the *grandfather_of* relation, This property enables the derivation of indirect relations through composition.

Type		Template
SYM	TT	{A} is the {R} of {B}.
	Q1	{A} is the {R} of ___?
	Q2	Who is the {R} of {A} ?
	Q3	{B} is the {R} of ___?
	Q4	Who is the {R} of {B} ?
ASYM	TT	{A} is the {R} of {B}.
	Q1	{A} is the {R} of ___?
	Q2	{B} is the {R'} of ___?
O2M	TT	{A} is the {R} of {B1}. {A} is the {R} of {B2}.
	Q1	{A} is the {R} of ___?
	Q2	{B1} and {B2} are the {R'} of ___?
TRANS	TT	{A} is the {R1} of {B}. {B} is the {R2} of {C}.
	Q1	{A} is the {R3} of ___?
	Q2	{C} is the {R3'} of ___?

Table 1: Templates for constructing the training data (TT) and evaluation (Q1-Q4).

3.2 Evaluation Setups

Datasets. We follow Table 1 to construct the datasets for training and evaluation. Specifically, for each type, we selected 10 relations from the widely used knowledge base Freebase (Bollacker et al., 2008) and instantiated 200 templates for each relation by randomly sampling from a person name list[1]. We then used ChatGPT to rephrase the training data into well-formatted sentences, while strictly preserving the order of entities and relations to mitigate potential issues related to autoregressive learning (Berglund et al., 2023). After training, we tested the model on Q1–Q4 to evaluate knowledge generalization.

Please refer to Appendix A for more details on dataset construction, including relation selection and quality control.

Learning Paradigms and LLM Backbones. We considered four learning paradigms:(i) **Supervised fine-tuning (SFT).** This is the standard approach for training LLMs on textual data. Here, we employed full-parameter training.(ii) **Retrieval-augmented generation (RAG).** In this paradigm, the model first retrieves evidence sentences from all training sentences, and then concatenates them with the query for inference.(iii) **In-context learning (ICL).** ICL can be seen as RAG with golden evidence. Here, we augment the prompt with few-shot examples to ensure the answer follows a spe-

[1]The US Baby Names dataset from Kaggle

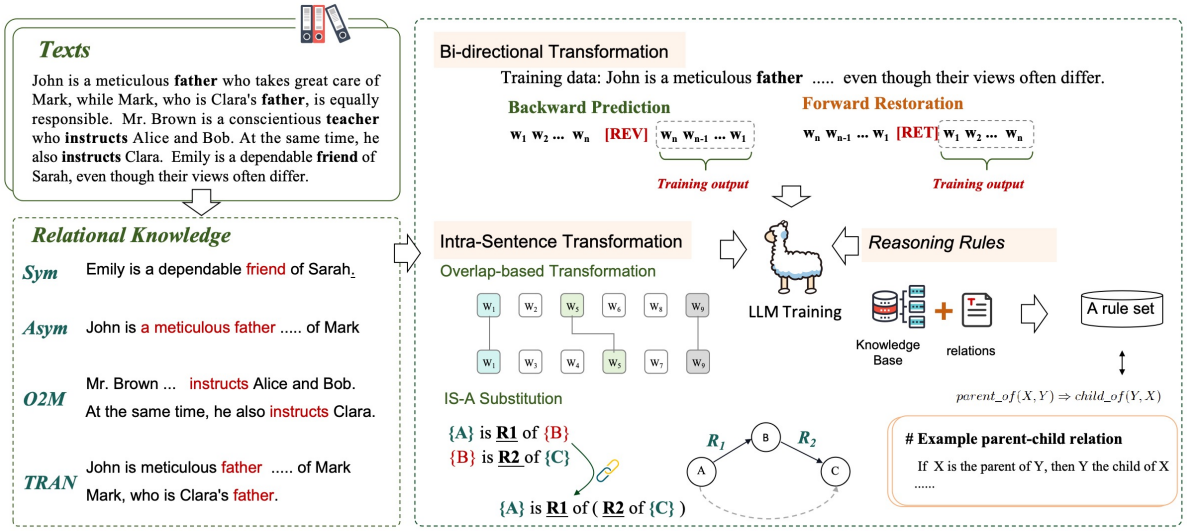

Figure 2: An overview of our approach, where we propose structural transformation noise (e.g., bidirectional and intra-sentence transformations) and explicit reasoning rules to enhance learning.

cific format for parsing.(iv) **Chain-of-thought reasoning (CoT).** CoT enhances prediction by encouraging step-by-step reasoning. We augment the prompt by adding the instruction, "Please think step by step".

For LLM backbones, we consider LLama3-8B, LLama3.2-1B (Dubey et al., 2024), and Mistral-7B (Jiang et al., 2023) to ensure diversity in parameter scale and architecture. We report 5-run average.

4 Learning with Transformation Noises and Reasoning Rules

A key issue behind LLMs' weak knowledge generalization is their rigid auto-regressive objective (Berglund et al., 2023). To address this, we propose (1) structural transformation noise, including bidirectional and advanced intra-sentence transformations, and (2) the incorporation of explicit reasoning rules to enhance learning.

4.1 Bi-directional Transformations

Let a training sentence be $\mathbf{x} = [w_1, \ldots, w_n]$, where w_i is the i^{th} word. We first set up a bidirectional transformation objective to either predict the reversed representation of $\mathbf{x}$, denoted as $\mathbf{x}' = [w_n, \ldots, w_1]$, or recover $\mathbf{x}$ from $\mathbf{x}'$. We design two independent supervised fine-tuning tasks, as shown below.

Backward Prediction. In this backward prediction task, the goal is to predict $\mathbf{x}'$ conditioned on $\mathbf{x}$ with a special instruction token [REV]. The input is denoted as $I = \mathbf{x} \oplus$ [REV]. Then, at time step t, the probability to be modeled is:

$$p_\theta^{(\mathrm{REV})}(t) = p_\theta\left(w_{n-t+1} \middle| I, w_n, \ldots, w_{n-t+2}\right) \tag{5}$$

where θ represents the model parameters. The training loss for the entire sentence can be defined as:

$$\mathcal{L}_{\mathrm{REV}} = -\sum_{t=1}^{n} \log p_\theta^{(\mathrm{REV})}(t) \tag{6}$$

In practice, we achieve the training by constructing a sequence $[\mathbf{x} \oplus [\mathrm{REV}] \oplus \mathbf{x}']$ and performing standard next-word prediction.

Forward Restoration. In this forward restoration task, the goal is to reconstruct the original sequence $\mathbf{x}$, conditioned on the reversed representation $\mathbf{x}'$ and an instruction token [RET], denoted by $I' = \mathbf{x}' \oplus$ [RET]. Similar to the previous task, the probability to model at time step t and the training loss are:

$$p_\theta^{(\mathrm{RET})}(t) = p_\theta\left(w_t \middle| X', w_1, \ldots, w_{t-1}\right) \tag{7}$$

$$\mathcal{L}_{\mathrm{RET}} = -\sum_{t=1}^{n} \log p_\theta^{(\mathrm{RET})}(t) \tag{8}$$

In practice, the training can also be achieved by constructing a sequence $[\mathbf{x}' \oplus [\mathrm{RET}] \oplus \mathbf{x}]$. Here, the introduction of instruction tokens [REV] and [RET], along with the two learning tasks, enables the model to learn bidirectional context.

4.2 Intra-Sentence Transformation

The bi-directional transformation is mainly for single training sentences. Here, we also consider intra-sentence transformation.

Overlap-based Transformation. The overlap-based transformation aims to merge the common parts shared by two sentences and concatenates the in-between parts to generate new training samples. For instance, consider the example in Figure 3. In this case, w_1, w_5, and w_9 are common subsequences (or more precisely, tokens). Here, we merge two sentences based on these tokens and concatenate the discontinuous parts in between to build a new sequence: [$\boxed{w_1}, w_2, w_3, w_4, \boxed{w_5}, w_6, w_8, w_7, \boxed{w_9}$]. Intuitively, this transformation helps the model learn which tokens tend to appear in a common context, such as those on the right-hand side of one-to-many relationships. In practice, we use the Longest Common Subsequence (LCS) algorithm to identify common subsequences. With dynamic programming, this results in a highly efficient algorithm with a time complexity of $O(N^2)$, where N is proportional to the length of the sentence.

IS-A Substitution. The motivation for *is-a substitution* is to identify semantically equivalent parts between sentences and generate new training examples by substituting these parts. Consider two training examples: 1) A is the $R1$ of B; 2) B is the $R2$ of C. We generate a new example: A is the $R1$ of (the $R2$ of C), where B serves as a proxy for substitution and is resolved across the two sentences. Through this approach, we aim to achieve the simplest chain based prediction. Note that we use only text matching and avoid complex parsing tools to ensure a simple and generalizable strategy.

4.3 Learning with Explicit Reasoning Rules

Different from format transformations, we also propose directly injecting reasoning rules into LLMs to improve generalization. To achieve this, we construct a rule set expressed in first-order logic (FOL) to capture the properties of different relation types (§ 3.1). For example, the FOL rule for the anti-symmetric relation (*parent_of*, *child_of*) is:

$$parent\_of(X, Y) \Rightarrow child\_of(Y, X) \quad (9)$$

Here, the rule is translated into plain text symbols for training, and we only instantiate the relations—not the entities—to eliminate the possibility of learning shortcuts. We train these reasoning rules separately from the relation texts to avoid inter-effects between them.

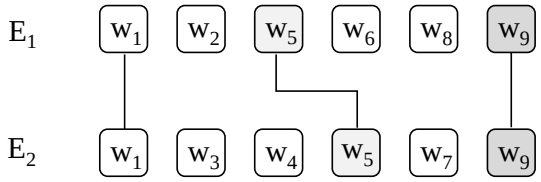

Figure 3: Example of the overlap patterns of two training examples E1 and E2.

Methods	Q1	Q2	Q3	Q4
				LLama3-8B
SFT	0.888	0.580	0.000	0.000
RAG	0.820	0.612	0.384	0.554
Ours	**0.900**	0.782	0.892	0.688
ICL	0.850	0.910	0.936	0.854
ICL + CoT	0.882	**0.922**	**0.986**	**0.886**
				LLama3.2-1B
SFT	0.566	0.410	0.000	0.000
RAG	0.438	0.628	0.346	0.636
Ours	**0.902**	**0.748**	0.562	0.648
ICL	0.552	0.750	0.760	0.748
ICL + CoT	0.682	0.738	**0.782**	**0.788**
				Mistral-7B
SFT	0.720	0.450	0.000	0.000
RAG	0.228	0.468	0.320	0.488
Ours	**0.982**	**0.830**	0.942	0.856
ICL	0.344	0.698	0.950	0.942
ICL + CoT	0.684	0.760	**0.958**	**0.962**

Table 2: Results for addressing symmetric relations.

5 Evaluation Results

In this section, we give a detailed analysis of the results on LLMs learning and generalizing across four types of relational knowledge.

5.1 Results on SYM relations

Table 2 presents the results on addressing symmetric relations, where Q1 and Q2 are settings with queries in the forward direction, and Q3 and Q4 are settings in the reverse direction. Here, Q2 and Q4 adopt queries with more natural expressions but with different word orders compared to Q1 and Q3 (see § 3.2 for details).

According to the results: (1) The standard SFT achieves good performance only in Q1, where the word orders of the query and the training examples match; it struggles with unmatched ones (Q2). Additionally, SFT fails to generalize symmetric re-

Methods	Q1	Q2	Avg.
			LLama3-8B
SFT	0.754	0.000	0.377
RAG	0.714	0.510	0.726
Ours	0.958	0.738	0.734
ICL	0.864	0.908	0.886
ICL + CoT	**0.964**	**0.988**	**0.976**
			LLama3.2-1B
SFT	0.882	0.000	0.441
RAG	0.768	0.718	0.743
Ours	0.956	**0.786**	**0.871**
ICL	0.888	0.878	0.883
ICL + CoT	**0.980**	0.008	0.494
			Mistral-7B
SFT	0.416	0.000	0.208
RAG	0.406	0.308	0.449
Ours	0.590	0.714	0.560
ICL	0.546	0.904	0.725
ICL + CoT	**0.648**	**0.980**	**0.814**

Table 3: Results for addressing anti-symmetric relations.

Methods	$Q1_{B1}$	$Q1_{B2}$	$Q1_{Both}$	Q2
				LLama3-8B
SFT	0.784	0.010	0.000	0.000
RAG	0.339	0.497	0.017	0.126
Ours	**0.930**	**0.810**	**0.748**	**0.416**
ICL	0.488	0.652	0.188	0.280
ICL + CoT	0.608	0.782	0.564	0.688
				LLama3.2-1B
SFT	0.324	0.012	0.000	0.000
RAG	0.042	0.292	0.000	0.179
Ours	**0.770**	0.488	0.328	0.326
ICL	0.230	0.422	0.046	0.230
ICL + CoT	0.438	**0.678**	**0.686**	**0.488**
				Mistral-7B
SFT	0.798	0.004	0.000	0.000
RAG	0.595	0.161	0.026	0.110
Ours	**0.980**	**0.936**	**0.918**	0.268
ICL	0.762	0.290	0.128	0.246
ICL + CoT	0.868	0.560	0.588	**0.556**

Table 4: Results for addressing one-to-many relations.

lations, achieving zero accuracy in Q3 and Q4. (2) In contrast, ICL, though adopting a "no learning" paradigm, still yields relatively good performance, and CoT further enhances learning. This indicates that LLMs may be better at leveraging evidence provided in the context. However, as expected, the more realistic setting RAG yields worse performance as it cannot retrieve golden evidence. (3) Our approach significantly outperforms SFT and RAG, and its strong performance in the reverse direction demonstrates its effectiveness in learning and generalizing symmetric relation knowledge.

5.2 Results on ASYM relations

Table 3 presents the results for addressing anti-symmetric relations, where Q1 and Q2 are settings with queries for the forward and reverse direction. We also report the average performance.

According to the results, SFT still cannot perform reasoning in the reverse direction, showing a lack of generalization. In contrast, our approach significantly outperforms SFT, especially in the reverse direction. This demonstrates the ability of our approach to handle anti-asymmetric relational knowledge. However, it is worth noting that the performance in the reverse direction is lower than in the forward direction. A possible reason is that for a relation R, its complementary relation R' may never appear in the training data, and the model must learn it from other knowledge, making it more challenging. Interestingly, we observe that ICL-like approaches do not exhibit this issue. This suggests that when LLMs reason from context, they naturally generalize the content. For the same reason, our method lags behind RAG on several metrics, particularly in the reverse direction.

5.3 Results on O2M relations

Table 4 presents the results on addressing one-to-many relations. Here, Q1 and Q2 denote the settings for querying in the forward and backward directions respectively. However, considering that the forward direction involves multiple answers, we further divide Q1 into the following cases: whether B1 is correctly answered ($Q1_{B1}$), whether B2 is correctly answered ($Q1_{B2}$), or whether both are correctly answered ($Q1_{Both}$).

Our approach significantly outperforms SFT on both $Q1_{Both}$ and Q2, highlighting its strength in generalizing one-to-many relational knowledge. Two key observations emerge when comparing other learning paradigms: (1) ICL-like methods perform poorly, especially on $Q1_{Both}$, likely due to confusion caused by one-to-many relations that may resemble conflicting knowledge. (2) We also observe architectural differences: LLaMA models prefer earlier entities (higher $Q1_{B1}$), while Mistral favors later ones (higher $Q1_{B2}$), suggesting distinct inductive biases in how evidence is processed.

Methods	Q1	Q2	Avg.
			LLama3-8B
SFT	0.004	0.000	0.002
RAG	0.232	0.046	0.139
Ours	**0.720**	0.424	**0.572**
ICL	0.332	0.146	0.239
ICL + CoT	0.488	**0.466**	0.477
			LLama3.2-1B
SFT	0.002	0.000	0.001
RAG	0.420	0.080	0.250
Ours	**0.984**	**0.570**	**0.777**
ICL	0.520	0.280	0.400
ICL + CoT	0.688	0.480	0.584
			Mistral-7B
SFT	0.000	0.000	0.000
RAG	0.380	0.000	0.190
Ours	0.538	**0.706**	**0.622**
ICL	0.480	0.200	0.340
ICL + CoT	**0.638**	0.484	0.561

Table 5: Results for addressing transitive relations.

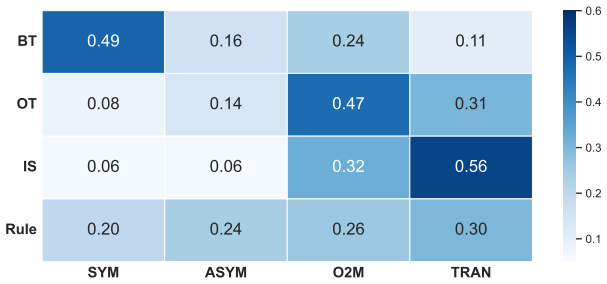

Figure 4: Ablation study on module impact across relation types. BT: bidirectional transformation; OT: overlap-based transformation; IS: IS-A substitution; Rule: reasoning rules.

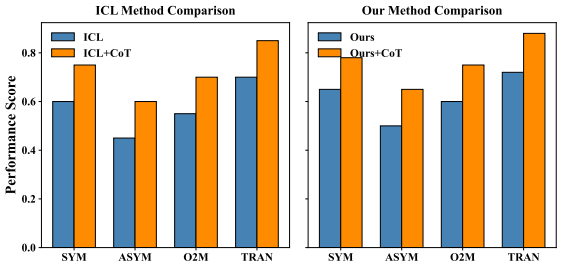

Figure 5: Effect of adopting CoT.

5.4 Results on TRAN relations

Table 5 presents the results for transitive relation tasks, where Q1 denotes forward compositional reasoning and Q2 denotes the more challenging backward direction. Overall, this setting is significantly difficult.The standard SFT method fails in both directions, showing limited generalization ability. ICL-based methods also perform poorly, especially on Q2, with accuracy below 0.3. Adding CoT improves results to around 0.5.In contrast, our method consistently outperforms others in both directions, demonstrating strong capability in handling compositional transitive knowledge. Among model backbones, Mistral performs best overall, and the strong performance of the LLaMA-1B model suggests that medium-sized LLMs can be effective for transitive reasoning.

6 Discussion

We provide a series of qualitative analysis to better understand the effectiveness of our approach.

6.1 Ablation Study

Different model components influence relation types differently. We conducted an ablation study using Llama-8B (Figure 4), where improvements from each component were normalized as relative scores.Results show that: (1) Bi-directional transformation (BT) benefits symmetric relations (SYM) by modeling reversibility, but is less effective on ASYM ones due to their directional nature. (2) Overlap-based transformation (OT) excels in O2M relations, likely due to its ability to model shared contexts. (3) IS-A substitution (IS) is most impactful for TRAN relations by constructing compositional structures. (4) Explicit reasoning rules help across all relation types, with especially strong effects on transitive relations.

6.2 Effect of Adopting Reasoning Rules

We conduct a deeper exploration of adopting reasoning rules by examining whether they can improve SFT and ICL. For SFT, we consider two settings: one follows our original approach, where reasoning rules are trained separately, and the other involves concatenating relevant rules with each training example for model training. The results are shown in Table 7, where we use LLaMA-3B and transitive relations for evaluation. Accordingly, incorporating rules improves performance in both SFT and ICL, suggesting their effectiveness. Moreover, in SFT, adding relevant rules for each training sentence is more effective, though this is not a realistic setting for real-world training.

Case	Train Exp.	Query	Answer	SFT	ICL	ICL + CoT	Ours	Ours + CoT
1	...John is the father...	Who is John's son?	Michael	William	... John's son is Michael.	John is the father of Michael. Therefore, Michael is John's son.	Michael	Michael. Michael is John's son...
2	...Sam is the teacher of Emily....of Michael...	Who are the students of Sam?	Emily and Michael	Emily	... the student of Sam is Michael	...the students of Sam are Emily and Michael.	...Emily and the student is Michael...	... students are Emily and Michael...
3	...John is Mark's brother ...uncle....	Who is the nephew of John?	Tom	James	Tom is John's nephew...	John is Mark's brother ... so Tom is John's nephew.	Elizabeth is the...	Tom. Tom is John's.....

Table 6: A case study comparing the outputs of different learning paradigms, where red represents incorrect predictions, and green represents correct predictions.

Methods	Q1	Q2	Avg
SFT	0.004	0.000	0.002
+ Sep. Training	0.212	0.112	0.162
+ Relev. Rules	0.244	0.248	0.228
ICL	0.332	0.146	0.239
+ Relev. Rules	0.438	0.256	0.396

Table 7: Effect of learning with reasoning rules.

Figure 6: Attention attribution analysis on anti-symmetric relations for LLaMA3-8B.

6.3 Effect of Adopting CoT

In Figure 5, we measure the average absolute improvement achieved by incorporating CoT into the prediction process. The results show that CoT is universally effective and compatible with both ICL and our method. This suggests that CoT may serve as a "free lunch" for relational reasoning.

6.4 Attribution Analysis

Here, we conduct an attribution analysis on the Llama3-8B model under anti-symmetric relations, examining how attention is distributed across each word in the input when generating the predefined correct answer "Tom." The results show that both the base model and the SFT model appear confused, failing to focus on informative parts of the input. In contrast, our model assigns higher attention to the relevant entity "John," and under the ICL setting, the model attends to both the relational cue "nephew" and the entity "John." This suggests that improved attention allocation plays a crucial role in helping the model adapt its output expression correctly.

6.5 Case study

In Table 6, we show specific cases for analysis. In Case 1, due to a lack of explicit contextual clues, SFT generates wrong answers, while both ICL and our method generate correct answers. In Case 2, ICL recognizes only "Michael" but misses "Emily" as the answer. By introducing CoT, the model generates a complete answer. Our method identifies both entities but delivers redundant expressions. When combined with CoT, it generates a clean output. In Case 3, for the compositional query, both SFT and our method fail to answer correctly. However, with a brief CoT prompt, our model can figure out the answer. This suggests that CoT activates latent reasoning capabilities in LLMs.

7 Conclusion

In this study, we evaluate LLMs' ability to learn and generalize relational knowledge—focusing on symmetric, antisymmetric, one-to-many, and transitive relations—and identify a clear gap between supervised fine-tuning and in-context learning. To address this, we propose a method that integrates transformation noise and logical rules, substantially improving model robustness and generalization. Our work provides both theoretical insights and practical strategies for enhancing LLMs in real-world knowledge-driven tasks.

Limitations

Our work has two primary limitations: (1) The data used is synthetic rather than real-world, while real-world data is more complex and can involve mixed properties, making it challenging to study and analyze their impact. (2) Empirical validation is currently limited to relations from Freebase, primarily focusing on person-related entities. Future work should test a broader range of entity types, including non-personal entities, to assess generalizability. Additionally, exploring theoretical explanations and establishing cross-domain evaluation benchmarks will be crucial to expand the applicability of our work.

References

Rami Aly, Marek Strong, and Andreas Vlachos. 2023. Qa-natver: Question answering for natural logic-based fact verification. *arXiv preprint arXiv:2310.14198*.

Gilbert Badaro, Mohammed Saeed, and Paolo Papotti. 2023. Transformers for tabular data representation: A survey of models and applications. *Transactions of the Association for Computational Linguistics*, 11:227–249.

Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. 2023. The reversal curse: Llms trained on" a is b" fail to learn" b is a". *arXiv preprint arXiv:2309.12288*.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, page 1247–1250, New York, NY, USA. Association for Computing Machinery.

Banghao Chen, Zhaofeng Zhang, Nicolas Langrené, and Shengxin Zhu. 2023. Unleashing the potential of prompt engineering in large language models: a comprehensive review. *arXiv preprint arXiv:2310.14735*.

Qiguang Chen, Libo Qin, Jiaqi Wang, Jinxuan Zhou, and Wanxiang Che. 2024a. Unlocking the capabilities of thought: A reasoning boundary framework to quantify and optimize chain-of-thought. *arXiv preprint arXiv:2410.05695*.

Zhiyuan Chen, Yaning Li, and Kairui Wang. 2024b. Optimizing reasoning abilities in large language models: A step-by-step approach. *Authorea Preprints*.

Liam Cripwell, Joël Legrand, and Claire Gardent. 2022. Controllable sentence simplification via operation classification. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2091–2103.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2021. Knowledge neurons in pretrained transformers. *arXiv preprint arXiv:2104.08696*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Yong Guan, Shaoru Guo, Ru Li, Xiaoli Li, and Hongye Tan. 2021. Frame semantic-enhanced sentence modeling for sentence-level extractive text summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4045–4052.

Haoyu Han, Yaochen Xie, Hui Liu, Xianfeng Tang, Sreyashi Nag, William Headden, Yang Li, Chen Luo, Shuiwang Ji, Qi He, et al. 2025. Reasoning with graphs: Structuring implicit knowledge to enhance llms reasoning. *arXiv preprint arXiv:2501.07845*.

Jiuzhou Han, Nigel Collier, Wray Buntine, and Ehsan Shareghi. 2023. Pive: Prompting with iterative verification improving graph-based generative capability of llms. *arXiv preprint arXiv:2305.12392*.

Myeongjun Erik Jang and Thomas Lukasiewicz. 2023. Consistency analysis of chatgpt. *arXiv preprint arXiv:2303.06273*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Yichen Jiang and Mohit Bansal. 2019. Self-assembling modular networks for interpretable multi-hop reasoning. *arXiv preprint arXiv:1909.05803*.

Tianjie Ju, Weiwei Sun, Wei Du, Xinwei Yuan, Zhaochun Ren, and Gongshen Liu. 2024. How large language models encode context knowledge? a layer-wise probing study. *arXiv preprint arXiv:2402.16061*.

Nora Kassner, Oyvind Tafjord, Ashish Sabharwal, Kyle Richardson, Hinrich Schuetze, and Peter Clark. 2023. Language models with rationality. *arXiv preprint arXiv:2305.14250*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E.

Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Zhe Lin, Yitao Cai, and Xiaojun Wan. 2021. Towards document-level paraphrase generation with sentence rewriting and reordering. *arXiv preprint arXiv:2109.07095*.

Hanmeng Liu, Ruoxi Ning, Zhiyang Teng, Jian Liu, Qiji Zhou, and Yue Zhang. 2023. Evaluating the logical reasoning ability of chatgpt and gpt-4. *arXiv preprint arXiv:2304.03439*.

Linqing Liu, Huan Wang, Jimmy Lin, Richard Socher, and Caiming Xiong. 2019. Mkd: a multi-task knowledge distillation approach for pretrained language models. *arXiv preprint arXiv:1911.03588*.

Eran Malach. 2023. Auto-regressive next-token predictors are universal learners. *arXiv preprint arXiv:2309.06979*.

Houman Mehrafarin, Arash Eshghi, and Ioannis Konstas. 2024. Reasoning or a semblance of it? a diagnostic study of transitive reasoning in llms. *arXiv preprint arXiv:2410.20200*.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*.

Eric Mitchell, Joseph J Noh, Siyan Li, William S Armstrong, Ananth Agarwal, Patrick Liu, Chelsea Finn, and Christopher D Manning. 2022. Enhancing self-consistency and performance of pre-trained language models through natural language inference. *arXiv preprint arXiv:2211.11875*.

Vishwas Mruthyunjaya, Pouya Pezeshkpour, Estevam Hruschka, and Nikita Bhutani. 2023. Rethinking language models as symbolic knowledge graphs. *arXiv preprint arXiv:2308.13676*.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, Kashun Shum, Randy Zhong, Juntong Song, and Tong Zhang. 2023. Ragtruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. *arXiv preprint arXiv:2401.00396*.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Autoosa Salari, Marco A Navarro, Mirela Milescu, and Lorin S Milescu. 2018. Estimating kinetic mechanisms with prior knowledge i: Linear parameter constraints. *Journal of General Physiology*, 150(2):323–338.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. 2023. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint arXiv:1902.10197*.

Armin Toroghi, Willis Guo, Ali Pesaranghader, and Scott Sanner. 2024. Verifiable, debuggable, and repairable commonsense logical reasoning via llm-based theory resolution. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6634–6652.

Chenguang Wang, Xiao Liu, and Dawn Song. 2020a. Language models are open knowledge graphs. *arXiv preprint arXiv:2010.11967*.

Jianing Wang, Qiushi Sun, Xiang Li, and Ming Gao. 2023a. Boosting language models reasoning with chain-of-knowledge prompting. *arXiv preprint arXiv:2306.06427*.

Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE transactions on knowledge and data engineering*, 29(12):2724–2743.

Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Guihong Cao, Daxin Jiang, Ming Zhou, et al. 2020b. K-adapter: Infusing knowledge into pre-trained models with adapters. *arXiv preprint arXiv:2002.01808*.

Siyuan Wang, Zhongyu Wei, Jiarong Xu, Taishan Li, and Zhihao Fan. 2023b. Unifying structure reasoning and language model pre-training for complex reasoning. *arXiv preprint arXiv:2301.08913*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Jundong Xu, Hao Fei, Liangming Pan, Qian Liu, Mong-Li Lee, and Wynne Hsu. 2024a. Faithful logical reasoning via symbolic chain-of-thought. *arXiv preprint arXiv:2405.18357*.

Zhuoyan Xu, Zhenmei Shi, and Yingyu Liang. 2024b. Do large language models have compositional ability? an investigation into limitations and scalability. *arXiv preprint arXiv:2407.15720*.

Jianxi Yang, Xiaoxia Yang, Ren Li, Mengting Luo, Shixin Jiang, Yue Zhang, and Di Wang. 2023. Bert and hierarchical cross attention-based question answering over bridge inspection knowledge graph. *Expert Systems with Applications*, 233:120896.

Michihiro Yasunaga, Antoine Bosselut, Hongyu Ren, Xikun Zhang, Christopher D Manning, Percy S Liang, and Jure Leskovec. 2022. Deep bidirectional language-knowledge graph pretraining. *Advances in Neural Information Processing Systems*, 35:37309–37323.

Wei Jie Yeo, Ranjan Satapathy, Rick Siow Mong Goh, and Erik Cambria. 2024. How interpretable are reasoning explanations from prompting large language models? *arXiv preprint arXiv:2402.11863*.

Zhao Zhang, Fuzhen Zhuang, Meng Qu, Fen Lin, and Qing He. 2018. Knowledge graph embedding with hierarchical relation structure. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3198–3207.

Hongyu Zhao, Kangrui Wang, Mo Yu, and Hongyuan Mei. 2023. Explicit planning helps language models in logical reasoning. *arXiv preprint arXiv:2303.15714*.

Yafang Zheng, Lei Lin, Shuangtao Li, Yuxuan Yuan, Zhaohong Lai, Shan Liu, Biao Fu, Yidong Chen, and Xiaodong Shi. 2024. Layer-wise representation fusion for compositional generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19706–19714.

Hanlin Zhu, Baihe Huang, Shaolun Zhang, Michael Jordan, Jiantao Jiao, Yuandong Tian, and Stuart Russell. 2024. Towards a theoretical understanding of the'reversal curse'via training dynamics. *arXiv preprint arXiv:2405.04669*.

A Dataset Construction

In this section, we describe how we construct the training and test sets for our Relational Reasoning Evaluation Dataset. Our goal is to ensure diversity in entity names, linguistic templates, and relation instantiations while preventing the model from memorizing the test data in advance, thereby rigorously assessing its generalization ability in learning relational structures.

A.1 Training Data Construction

Defining Common Relationships. We select ten frequently used relationships for each of the four relation types described in Table 8. These are drawn from common social or familial roles and from typical hierarchical scenarios.

Entity Names and Privacy. To ensure diversity and fairness in the training data, we balanced male and female names in the original dataset, which were drawn from culturally neutral common name lists. Additionally, we avoided using any personally identifiable information or widely recognized public figures during the selection process to maintain anonymity.

Constructing Samples. For each relation type, we use multiple statement templates and corresponding question forms (see Table 1). Each template is instantiated with the selected relationships (e.g., "Alice is the friend of Bob"), thereby constructing our training data samples. Within the same relational experiment, each name corresponds to a unique relationship.

Quality Control. We perform grammar and syntax checks on all generated samples and ensure the consistency of entity names and gender-related relationships to maintain semantic coherence. Additionally, we remove ambiguous or contradictory sample pairs to enhance data quality.

A.2 Test Data Construction

Constructing Samples. According to the rules, we generate corresponding test data for each training sample. By using the same entity names and corresponding templates, we instantiate each template with the selected relationships to construct our test dataset. **Pre-Training Test for Zero-Shot Accuracy.** Before training, we perform a zero-shot accuracy test on the model using the test dataset to ensure it cannot answer correctly without training. The goal is to verify that the model's baseline performance is low enough so that any improvements after training are due to actual learning rather than pre-existing knowledge. Our experiments confirm that the accuracy on all test data is 0 with the pretrained model.

Symmetric Relations
friend, classmate, neighbor, colleague, roommate, teammate, partner, ally, cofounder, sibling
Anti-Symmetric Property
father-son, teacher-student, mother-daughter, grandfather-grandson, husband-wife, uncle-nephew, aunt-niece, boss-employee, doctor-patient, nurse-patient
One-to-Many Property
coach-athlete, doctor-patient, nurse-patient, landlord-tenant, buyer-seller, client-consultant, host-guest, mentor-mentee, driver-passenger, editor-writer
Transitive Property
grandparent-parent-child, sibling-sibling-child, mentor-mentee-mentee, coach-athlete-athlete, doctor-patient-patient, landlord-tenant-tenant, buyer-seller-seller, host-guest-guest, uncle-nephew-child, aunt-niece-child

Table 8: The detailed sub-relation tables for the four relation types illustrate common real-world relational structures. Through subsequent relation definitions and reasoning processes, they reveal different logical characteristics of relation inference.

B Details of Reasoning-Rules

In this section, we present a detailed explanation of the process used to construct our rule-based reasoning data. Our data is derived from the Wikidata database, a structured, open-source knowledge graph that contains entities and their relationships across diverse domains.We leverage this resource to extract relationships and build a comprehensive dataset of reasoning rules.

We adopt different strategies for various relationship categories. For symmetric property, we search for property descriptions that include the term symmetric. For antisymmetric property, we look for descriptions containing the term "inverse" and manually pair inverse properties by consulting the Inverse Property field (e.g., father $\leftrightarrow$ child). For one-to-many property, we select those with an ob-

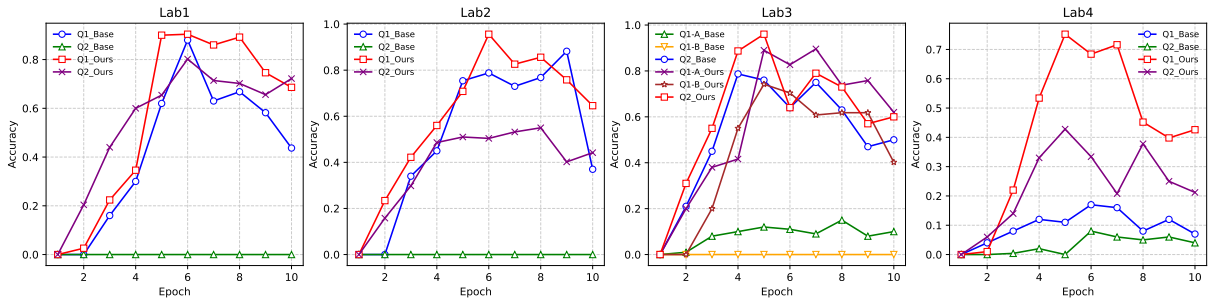

Figure 7: Model ACC changes with training epochs

ject cardinality specified as "multiple". Finally, for transitive property, we identify properties that support path expressions (e.g., ID/ID) or are explicitly designated as transitive.

For each relationship category, we determine the number of entities required for each target relation. Specific entities are replaced with a predefined set of placeholders (e.g., PersonA, Mr.B, Mrs.C), which facilitates substitution during sentence generation and enhances the model's generalization capabilities. The relationships are then represented as tuples (relation1,relation2, object1, object2, . . .); for example, (employer, employee,PersonA, PersonB,). By flexibly substituting these placeholders, we generate diverse sentence structures, further improve the model's understanding of inter-entity relationships during training.Additional. We construct the training data using predefined language structure templates. These templates follow a condition-result reasoning format, where the condition describes the relationship between entities and the result denotes the logical conclusion derived from that relationship. A sample of these templates is provided in Table 9.

For each template, we synthesize appropriate conditions and results that reflect the characteristics of the relationships. The substitution process entails selecting suitable predicates and articles for connection and modification. For instance, in the case of symmetric relations, we generate two semantically equivalent templates for the condition, such as "A is the R of B" and "The R of B is A." For each condition template, we iterate through various relationships within each category.Template slots are dynamically filled using a type-aware alignment module that maps entity and relation types to lexical patterns. This process generates a dataset of 200 explainable rules, each pairing a formal logic expression with its natural language equivalent, enabling joint training of large language models on both symbolic and textual representations of relational knowledge. To prevent repetition in instantiated relationships, we randomly select appropriate names to fill entity placeholders, completing the instantiation.

No.	Language Structure Templates
1	Given the [condition], then [result].
2	If [condition], then [result].
3	When [condition], then [result].
4	Once [condition] is met, then [result].
5	If [condition] holds true, then [result].
6	Provided that [condition], then [result].
7	In case [condition], then [result].
8	Assuming [condition], then [result].
9	If [condition] is satisfied, then [result].
10	When [condition] occurs, then [result].

Table 9: List of language structure templates with placeholders for conditions and results.

C Hyperparameters Setting

We fine-tune the base model using supervised learning. During training, we set the number of epochs to 5, the learning rate to 2e-5, and the batch size to 32 for the LLaMA3-8B and LLaMA3.2-1B models. For the Mistral-7B model, we set the number of epochs to 5, the learning rate to 4e-5, and the batch size to 16. During inference, we utilize the vLLM framework (Kwon et al., 2023) to efficiently handle large-scale model evaluations. All experiments are conducted on NVIDIA A100 GPUs with 80GB of memory.

D Epoch Analysis for Relational Understanding

In this section,we investigate the correlation between training epochs and the model's knowledge internalization. We employ a consistent forward-oriented training template and evaluate performance through both forward and backward reasoning metrics. The experimental results for four relational patterns are presented in Figure 7 .Our empirical evidence indicates that the model's forward reasoning capability demonstrates a statistically significant ascending trajectory with increasing training epochs, despite minor fluctuations observed in certain epochs. However, the model shows minimal improvement in backward tasks, one-to-many recognition, and multi-hop reasoning tasks. Our method achieves consistent performance improvements across all evaluation metrics, further reflecting the framework's effectiveness in facilitating comprehensive knowledge acquisition.