Assessing Logical Inference Capabilities of Large Language Models through RDF Schema Entailment Rules: A Multi-Level Evaluation

Taichi Hosokawa[1], Sudesna Chakraborty[1] and Takeshi Morita[1,2,*]

[1]*Aoyama Gakuin University, Kanagawa, Japan*
[2]*National Institute of Advanced Industrial Science and Technology, Tokyo, Japan*

Abstract

Large language models (LLMs) achieve strong performance in various language tasks, yet their logical inference abilities remain limited. LLMs often rely on pre-trained knowledge rather than explicit inference. Their inference capabilities in ontology languages like RDFS also remain underexplored. This study evaluates LLMs' inference abilities using RDFS entailment rules with two knowledge datasets: real-world data from Linked Open Data and counterfactual data created by systematically altering real-world facts. We propose a novel evaluation methodology assessing LLM outputs. To analyze inference behavior under different conditions, we design a three-level task framework varying rule presentation methods for identical inference tasks. Results show high accuracy on real-world datasets. LLMs sometimes infer missing premises using pre-trained knowledge, suggesting potential for incompletely structured environments. However, accuracy declines with counterfactual datasets and when shifting from pre-combined to multiple separate rules. Performance further drops when models must select appropriate rules from predefined subsets. These findings highlight both strengths and limitations of LLMs in structured, rule-based inference within ontology-driven systems.

Keywords

Large Language Models, RDF Schema entailment rules, logical inference capability, counterfactual knowledge

1. Introduction

In recent years, large language models (LLMs) have demonstrated substantial capabilities across natural language processing tasks, yet their capacity for logical inference remains under scrutiny [1]. LLMs tend to leverage pretrained knowledge rather than explicit logical inference, leading to evaluation studies under counterfactual conditions [2, 5]. While existing assessments using RDFS inference rules have relied on natural language-based methods, such approaches present challenges for rigorous verification [3]. To address this, we propose a novel evaluation framework that rigorously assesses LLM logical inference abilities using RDFS entailment rules. Our framework assesses LLM inference capabilities through real-world and counterfactual knowledge datasets, different rule presentation methods, and multi-rule combinations, using RDFS triples for precise, quantitative evaluation.

2. Related Work

Wu et al. [3] investigated whether pre-trained language models can store, understand, and utilize ontological knowledge for inference. They evaluated inference tasks based on six RDFS entailment rules using DBpedia [6] and Wikidata [7] data. However, significant limitations in inference accuracy for complex inference were revealed. Ozeki et al. [5] assessed syllogistic inference abilities of LLMs and showed performance decline in tasks involving belief-incongruent (counterfactual) premises and conclusions. Morishita et al. [2] assessed inference accuracy when constants and predicates were replaced with randomly assigned vocabulary, testing whether LLMs could apply logical rules without

ISWC 2025 Companion Volume, November 2–6, 2025, Nara, Japan

[*] Corresponding author.

sudesna@it.aoyama.ac.jp (S. Chakraborty); morita@it.aoyama.ac.jp (T. Morita)
0000-0002-3963-1761 (S. Chakraborty); 0000-0001-8963-2562 (T. Morita)

Table 1
Selected RDFS Entailment Rules

Rule	Premise	Conclusion

rdfs2	`<i, rdfs:domain, X>`
	`<a, i, b>`
		`<a, rdf:type, X>`

rdfs3	`<i, rdfs:range, X>`
	`<a, i, b>`
		`<b, rdf:type, X>`

rdfs5	`<i, rdfs:subPropertyOf, j>`
	`<j, rdfs:subPropertyOf, k>`
		`<i, rdfs:subPropertyOf, k>`

rdfs7	`<i, rdfs:subPropertyOf, j>`
	`<a, i, b>`
		`<a, j, b>`

rdfs9	`<X, rdfs:subClassOf, Y>`
	`<a, rdf:type, X>`
		`<a, rdf:type, Y>`

rdfs11	`<X, rdfs:subClassOf, Y>`
	`<Y, rdfs:subClassOf, Z>`
		`<X, rdfs:subClassOf, Z>`

relying on pre-trained knowledge. The present study is similar to Wu et al. in using RDFS entailment rules to evaluate LLMs, but employs RDFS triples as output format and evaluates inference tasks involving multiple rules, enabling more rigorous and quantitatively precise assessment.

3. Methodology

3.1. RDFS Entailment Rules

We select six RDFS entailment rules that are frequently used in practical Semantic Web settings (Table 1). These rules are well-suited for assessing LLM inference from a practical perspective. We further define 13 additional composite rules by combining two rules (rdfs2+3, rdfs2+7, rdfs2+9, rdfs3+7, rdfs3+9, rdfs5+7, rdfs9+11) and three rules (rdfs2+3+7, rdfs2+3+9, rdfs2+5+7, rdfs2+9+11, rdfs3+5+7, rdfs3+9+11), yielding 19 rules in total to evaluate inference patterns of varying complexity.

3.2. Evaluation Datasets

To systematically evaluate LLM inference capabilities, we conduct evaluations under diverse dataset conditions.

Real-world knowledge dataset (RK): Constructed from Linked Open Data sources including DBpedia [6], Wikidata [7], and Linked Open Vocabularies [8].

Counterfactual datasets: Eight variants created primarily by systematically modifying real-world knowledge data:

- **S**: Swaps subjects/objects, property domains/ranges, and reorders class/property hierarchies
- **NA/NR**: Adds “not” prefix to all/random classes and properties
- **SNA/SNR**: Combines S with NA/NR modifications
- **GS**: Shuffles all resources ensuring none remain in original positions
- **GSC**: Applies GS then renames resources using DBpedia’s type-appropriate naming conventions (PascalCase for classes, Upper_Snake_Case for instances, camelCase for properties)
- **RND**: Creates new triples using randomly assigned DBpedia vocabularies by resource type [2]

Random symbolic dataset (NS): Triples formed from 8-character alphanumeric strings with no semantic meaning.

3.3. Task Levels and Prompt Design

We define three task levels for a common inference task by varying how the inference rules are presented: The prompts provide inference rules and premise knowledge, instructing LLMs to perform inference

Table 2
Example of Composite Rule `rdfs2_3_7`

Rule	Premise	Conclusion

`rdfs2_3_7`	`<i, rdfs:domain, X>`
	`<i, rdfs:range, Y>`
	`<i, rdfs:subPropertyOf, j>`
	`<a, i, b>`
		`<a, rdf:type, X>`
		`<b, rdf:type, Y>`
		`<a, j, b>`

based solely on the given premises. Output formats are strictly specified to ensure consistent evaluation. Each task level using an example that combines `rdfs2`, `rdfs3`, and `rdfs7` as the target inference rules are shown in the following:

Lv1: Single composite rule provided; model applies it to premise knowledge. For example, `rdfs2_3_7` (Table 2):

Lv2: Only the necessary rules from Table 1 are presented separately (e.g., `rdfs2`, `rdfs3`, and `rdfs7`); model applies all given rules.

Lv3: Complete set of six rules (Table 1) provided; model must select relevant rules (e.g., `rdfs2`, `rdfs3`, and `rdfs7` from the six available), apply them, and report which rules were used.

Strict prompt templates define output formats to prevent background knowledge incorporation and proficient result extraction.

3.4. Evaluation Metrics

We extract inferred triples from model outputs and compare them with conclusions from an RDFS reasoner using Apache Jena Inference API [9]. To mitigate superficial variations, each triple component is matched using a 0.95 string-similarity threshold. Precision, recall, and F1 score are then computed. For Lv3, rule-name selection is further evaluated by exact string matching.

4. Experimental Results

We evaluated GPT-4o mini (gpt-4o-mini-2024-07-18) and GPT-4o (gpt-4o-2024-08-06) on 19 entailment rules across all datasets and task levels. For RK and counterfactual datasets (excluding RND), 400 samples per rule were used; 100 samples per rule were used for RND and NS.

4.1. Inference Performance

Table 3 shows average F1 scores by rule type (1-rule: single RDFS entailment rule, 2-rule: combinations of two rules, 3-rule: combinations of three rules) and task level. GPT-4o consistently outperformed GPT-4o mini across all conditions, maintaining high inference accuracy even under complex conditions (3-rule and Lv3). Both models showed decreased F1 scores as task complexity increased, with GS consistently recording the lowest scores, while GSC consistently outperforming GS. For GPT-4o mini, performance dropped significantly under 3-rule Lv3 conditions (F1=0.30 for GS, 0.38 for GSC), whereas GPT-4o maintained relatively high performance (F1=0.71 for GS, 0.82 for GSC under same conditions).

4.2. Rule Selection Performance

Table 4 shows average F1 scores for rule selection accuracy in Lv3 tasks. GPT-4o demonstrated high rule selection accuracy, with F1 scores above 0.89 for all datasets. In contrast, GPT-4o mini showed an unusual trend: lower rule selection accuracy in 1-rule tasks compared to 2-rule and 3-rule tasks. An analysis of error cases suggests that this counterintuitive trend stems from the model frequently inferring knowledge not present in the prompt, leading to the incorrect application of additional rules. To illustrate this behavior, Table 5 shows a case where GPT-4o mini incorrectly applied `rdfs3` in addition to the required `rdfs2`. Although no `rdfs:range` information was provided, the model supplemented it

Table 3
Average F1 scores by rule type and task level

Rule Type	Level	GPT-4o mini										GPT-4o									
		RK	S	NA	NR	SNA	SNR	GS	GSC	RND	NS	RK	S	NA	NR	SNA	SNR	GS	GSC	RND	NS
1-rule	Lv1	0.89	0.85	**0.91**	0.90	0.88	0.87	0.66	0.77	0.87	0.86	0.95	0.94	0.96	0.96	0.95	0.94	0.88	0.91	0.94	**0.97**
	Lv2	0.89	0.85	**0.91**	0.90	0.88	0.87	0.66	0.77	0.87	0.86	0.95	0.94	0.96	0.96	0.95	0.94	0.88	0.91	0.94	**0.97**
	Lv3	0.70	0.68	0.75	0.72	0.71	0.70	0.49	0.56	0.65	**0.77**	0.94	0.92	**0.95**	0.95	0.93	0.93	0.83	0.91	0.91	0.91
2-rule	Lv1	0.92	0.88	0.89	0.91	0.88	0.89	0.64	0.73	0.88	**0.93**	**0.95**	0.89	0.95	0.94	0.93	0.92	0.88	0.91	0.93	0.95
	Lv2	**0.83**	0.81	0.83	0.83	0.81	0.74	0.52	0.61	0.80	0.79	0.96	0.93	0.96	**0.97**	0.95	0.88	0.83	0.87	0.96	0.97
	Lv3	0.66	0.62	0.65	**0.68**	0.63	0.61	0.35	0.44	0.65	0.61	**0.92**	0.89	0.92	0.92	0.89	0.91	0.79	0.86	0.90	0.91
3-rule	Lv1	0.83	0.82	0.80	0.83	0.78	0.80	0.50	0.59	0.83	**0.84**	0.94	0.92	**0.96**	0.95	0.95	0.94	0.87	0.91	0.93	0.96
	Lv2	**0.72**	0.68	0.71	0.71	0.67	0.67	0.40	0.49	0.68	0.63	0.94	0.91	0.93	0.93	0.91	0.92	0.79	0.88	0.94	**0.96**
	Lv3	**0.64**	0.60	0.58	0.61	0.56	0.58	0.30	0.38	0.58	0.56	**0.89**	0.86	0.88	0.88	0.86	0.86	0.71	0.82	0.88	0.88

Table 4
Average F1 scores for rule selection in Lv3 tasks

Rule Type	GPT-4o mini										GPT-4o									
	RK	S	NA	NR	SNA	SNR	GS	GSC	RND	NS	RK	S	NA	NR	SNA	SNR	GS	GSC	RND	NS
1-rule	0.84	0.84	**0.85**	0.85	0.85	0.85	0.76	0.77	0.85	0.80	0.98	0.99	0.99	0.99	**1.00**	0.99	0.98	0.99	1.00	0.99
2-rule	0.89	0.90	0.92	0.92	0.90	0.86	0.81	0.82	0.95	**0.96**	0.99	0.98	0.99	0.99	0.99	0.92	0.89	0.91	0.99	**1.00**
3-rule	0.82	0.85	0.81	0.82	0.82	0.83	0.84	0.85	0.87	**0.91**	0.96	0.97	0.97	0.97	0.96	0.97	0.93	0.96	0.97	**0.98**

Table 5
Example of incorrect rule application: GPT-4o mini applies `rdfs3` in a task requiring only `rdfs2`.

Premise Knowledge

`<residence, rdfs:domain, Person>`
`<Camille_Leblanc-Bazinet, residence, Boulder,_Colorado>`

Expected Output

`<Camille_Leblanc-Bazinet, rdf:type, Person>`	`rdfs2`

Model Output

`<Camille_Leblanc-Bazinet, rdf:type, Person>`	`rdfs2`
`<Boulder,_Colorado, rdf:type, Residence>`	`rdfs3`

from context and produced an extra inference. Such overgeneralization lowers rule selection accuracy, especially under the 1-rule condition.

5. Discussion and Conclusion

Both models achieved high inference accuracy on real-world knowledge tasks. GPT-4o maintained stable F1 scores even in complex settings (3-rule, Lv3), demonstrating effective rule-based logical inference. GPT-4o consistently outperformed GPT-4o mini, suggesting that larger parameters and broader training data enable more complex reasoning.

However, GPT-4o mini tended to supplement missing knowledge even in tasks requiring strict rule-based inference. While logically inconsistent, this behavior may prove advantageous in real-world applications where knowledge incompleteness is unavoidable.

Inference accuracy was consistently lowest on the GS dataset across all conditions, with GSC showing relatively low but better performance than GS. This suggests LLMs rely on lexical cues and naming conventions in resource names as heuristics for inference. GPT-4o's results indicate that increased model scale can reduce this reliance and enhance structurally grounded inference.

These findings highlight both strengths and limitations of LLMs in ontological inference. While LLMs demonstrate strong potential for RDFS reasoning in realistic settings, their weaker performance under counterfactual conditions and reliance on linguistic patterns highlight critical areas for improvement, particularly in generalizing to unfamiliar data. Future work should focus on analyzing LLMs' inference processes in greater detail and extend evaluations with more expressive ontology languages such as OWL.

Supplemental Material: Evaluation code, datasets, and detailed results are available at https://anonymous.4open.science/r/LLM-InferRDFS-MultiLevelEval-EBFC/.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers 23K11221 and 25K03232, and was partially supported by NEDO under Grant Numbers JPNP20006 and JPNP25006.

Declaration on Generative AI

During the preparation of this work, the authors used OpenAI ChatGPT and Anthropic Claude in order to: Grammar and wording check, and translation support. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] Huang, J., Chang, K.C.C.: Towards reasoning in large language models: A survey. In: Findings of the Association for Computational Linguistics: ACL 2023, pp. 1049–1065 (2023)

[2] Morishita, T., Yamaguchi, A., Morio, G., Sogawa, Y.: JFLD: A Japanese benchmark for deductive reasoning based on formal logic. In: Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), pp. 3034–3049 (2024)

[3] Wu, W., Jiang, C., Jiang, Y., Xie, P., Tu, K.: Do PLMs Know and Understand Ontological Knowledge? In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 3080–3101. Association for Computational Linguistics (2023).

[4] Guha, R.V., Brickley, D., Macbeth, S.: Schema.org: evolution of structured data on the web. Communications of the ACM 59(2), 44–51 (2014)

[5] Ozeki, H., Oba, S., Mita, M., Hisamoto, S., Yoshinaga, N.: NeuBAROCO: A Japanese dataset for evaluation of syllogistic reasoning ability of language models. In: Proceedings of the 30th Annual Meeting of the Association for Natural Language Processing, pp. 1776–1781 (2024)

[6] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: DBpedia: A nucleus for a web of open data. In: The Semantic Web. ISWC 2007, ASWC 2007. LNCS, vol. 4825, pp. 722–735. Springer, Heidelberg (2007)

[7] Vrandečić, D., Krötzsch, M.: Wikidata: a free collaborative knowledgebase. Communications of the ACM 57(10), 78–85 (2014)

[8] Vandenbussche, P.Y., Atemezing, G.A., Poveda-Villalón, M., Vatant, B.: Linked Open Vocabularies (LOV): A gateway to reusable semantic vocabularies on the Web. Semantic Web 8(3), 437–452 (2017)

[9] Apache Software Foundation. Reasoners and rule engines: Jena inference support. https://jena.apache.org/documentation/inference/ (accessed July 2025)