Beyond Statistical Parroting: Hard-Coding Truth into LLMs via Ontologies

Hamed Babaei Giglou[1,7,*,†], Simon Burbach[2,7,*,†], Francesco Compagno[3,7,*,†], Muhammad Ismail[4,7,*,†], Gunjan Singh[5,7,*,†] and Sebastian Rudolph[6,7,*,†]

[1]TIB - Leibniz Information Centre for Science and Technology, Hannover, Germany
[2]Helmut Schmidt University, Hamburg, Germany
[3]University of Trento, Trento, Italy
[4]University of South Wales, Pontypridd, UK
[5]FIZ Karlsruhe, Karlsruhe, Germany
[6]TU Dresden, Dresden, Germany
[7]Team Jedi at Semantic Web Research Summer School 2025, Bertinoro, Italy

Abstract

Large Language Models (LLMs) are powerful but prone to hallucinations and factual inconsistencies, especially in knowledge-intensive tasks. In this work, we explore the integration of structured ontological knowledge into LLM prompts as a strategy to enhance factual accuracy and reliability. Using the Pizza Ontology as a showcase, we probe how different levels of domain grounding—ranging from base prompts to ontology-informed prompts—affect the factual accuracy of LLM responses. We tested different instruction-tuned last-generation LLMs from the Qwen and Llama families, ranging from 0.5 to 72 billion parameters, on a dataset of approximately 51 questions requiring various types of reasoning. Our results show that injecting ontological axioms into prompts improves response accuracy, demonstrating that formal domain knowledge can significantly reduce hallucinations. This proof-of-concept study highlights the potential for combining symbolic approaches with LLMs and lays the groundwork for more reliable, explainable AI systems. Our codebase is available at https://github.com/HamedBabaei/OntoTruth.

Keywords

Ontology Reasoning, Large Language Models, Reliable AI, Knowledge Retrieval

1. Introduction

It has often been claimed that knowledge graphs (KGs) such as Wikidata can serve as a kind of "ground truth" for AI systems, especially as large language models (LLMs) have become increasingly prone to generating hallucinated content [1, 2, 3]. While LLMs excel at generating fluent and coherent text, their outputs often suffer from factual inconsistencies and fabricated details. These issues are not mere imperfections; they pose serious risks to trust and reliability in high-stakes domains such as conservation planning, healthcare, and education, and more broadly hinder the use of LLMs for knowledge-intensive tasks.

To address these challenges, there is a growing interest in retrieval-augmented generation (RAG), where outputs from LLMs are guided or constrained by retrieved knowledge from external sources. Among such sources, ontologies and structured knowledge graphs offer a compelling advantage: they encode domain knowledge in a logic-based, verifiable manner. This makes them not only useful for guiding generation, but also for retrieving precise, context-aware information that can anchor LLM responses.

RAGE-KG 2025: The Second International Workshop on Retrieval-Augmented Generation Enabled by Knowledge Graphs, co-located with ISWC 2025, November 2–6, 2025, Nara, Japan

*Corresponding authors.
†These authors contributed equally.
hamed.babaei@tib.eu (H. Babaei Giglou); burbachs@hsu-hh.de (S. Burbach); francesco.compagno@unitn.it (F. Compagno); muhammad.ismail@southwales.ac.uk (M. Ismail); gunjan.singh@fiz-karlsruhe.de (G. Singh); sebastian.rudolph@tu-dresden.de (S. Rudolph)
0000-0003-3758-1454 (H. Babaei Giglou); 0009-0009-8373-6622 (S. Burbach); 0000-0003-1002-608X (F. Compagno); 0000-0003-3113-7416 (M. Ismail); 0000-0003-3171-9088 (G. Singh); 0000-0002-1609-2080 (S. Rudolph)

However, realizing this vision presents several key challenges. One central challenge is *efficient retrieval and injection*: how do we select and embed relevant ontology fragments into the LLM's context? Addressing this requires controlled experimentation to disentangle the effects of ontological grounding from the LLM's own latent knowledge. To explore the usefulness and feasibility of providing ontological knowledge through LLM prompts, our research has been guided by the following research questions:

- **RQ1.** To what extent can domain-specific ontological knowledge reduce hallucinations and factual errors in LLM outputs?
- **RQ2.** What are the practical challenges and limitations when providing ontological sources as part of an LLM prompt?

Given a question Q and a domain ontology $Onto$, our objective is to generate a response R that incorporates domain-specific knowledge retrieved from $Onto$, where $Onto$ includes its asserted axioms and may additionally be extended with reasoner-inferred knowledge. By doing so, we aim to ground LLM outputs in structured knowledge, thereby reducing hallucinations and improving transparency - both of which are critical for building more reliable AI systems. While preliminary and limited in scope, this study serves as an initial step toward future neuro-symbolic systems that combine the generalization ability of LLMs with the rigor of ontological reasoning. Specifically, this paper makes the following contributions:

- We introduce a proof-of-concept framework that grounds LLM outputs in structured domain ontologies via prompt injection.
- We design controlled experiments using the well-known Pizza Ontology (PO) to systematically evaluate how ontology-informed prompting affects hallucination reduction and factual consistency.
- We present a quantitative comparison of LLM responses across multiple categories of reasoning and ontology-materialized settings.
- We analyze both the benefits and the limitations of ontology-based grounding, offering a roadmap for future neuro-symbolic integration.

Our approach aims to improve the reliability, factual accuracy, and contextual relevance of LLM outputs by incorporating structured ontological knowledge across the input, reasoning, and output stages. In this setup, the ontology is treated as the primary source of truth, while the LLM's internal knowledge may be incomplete or inaccurate. Since we use a simple, common-sense ontology designed for instructional purposes, this assumption may not always hold — there can be cases where the LLM knows more than the ontology. However, in the case of complex ontologies that represent specialized domains, which are the main focus of our approach, we expect the ontology to offer a more complete and authoritative knowledge base.

The rest of the paper is organized as follows: Section 2 discusses related work, Section 3 presents our conceptual architecture for automatic LLM response generation grounded in ontological knowledge, Section 4 describes our experimental setup and results, and Section 5 concludes the paper. The work reported in this paper is based on a joint student research project at ISWS2025, the International Semantic Web Research Summer School.[1]

2. Related Work

Recently, there has been growing interest in combining ontologies with LLMs to join the strengths of both approaches. Several studies have explored neuro-symbolic integration frameworks that translate LLM-generated text into logical forms, which are then checked for consistency against ontological axioms. For instance, [4] proposes a pipeline using OWL ontologies and symbolic reasoners to detect inconsistencies in LLM-generated statements, providing iterative feedback to reduce hallucinations

[1] https://2025.semanticwebschool.org/

and enhance semantic coherence. Similarly, [5] improves question-answering systems by validating and repairing LLM-generated SPARQL queries using ontological constraints, resulting in significant accuracy gains in enterprise database contexts. Finally, [6] envisions a reasoning-based loop in which eventual inconsistencies of the LLM output with the ontology are found, verbalized, and used to attempt to repair the LLM output.

Beyond strict logical consistency, other studies investigate softening the rigidity of ontological reasoning by incorporating the flexibility of LLMs. [7] introduces a neurosymbolic approach where LLMs modulate rule activations in knowledge graphs, balancing robustness with adaptability in reasoning. Moreover, ontology-driven prompt tuning has been proposed to incorporate domain knowledge directly into LLM inputs. For example, [8] demonstrates improved task and motion planning in robotics by enriching prompts with ontology-based context and environment state descriptions, resulting in semantically valid and context-aware plans.

In domains involving viewpoint analysis, hybrid human-LLM frameworks use LLMs to identify and classify perspectives within news media, as shown by [9]. This approach leverages LLMs' linguistic capabilities guided by ontological categories to support scalable, automated content analysis. Investigating the domain adaptability of LLMs, [10] conduct controlled experiments revealing that off-the-shelf LLMs mainly rely on learned lexical senses rather than true semantic reasoning when extracting structured knowledge. However, fine-tuning LLMs on domain-specific data can improve their performance in ontology learning tasks involving arbitrary domain terms.

Such a fine-tuning may itself be driven by an ontology: for instance, [11] improves the performance of LLMs of evaluating sentence similarity by "infusing" knowledge from a biomedical ontology. This is done through contrastive learning over a dataset of couples of similar and dissimilar definitions of relevant medical concepts generated from the ontology with the help of another LLM. The work in [12] also employs a pipeline that uses an ontology and enhances LLM prediction of protein interactions. In this case, the ontology is utilized in two ways: the topology of the network of entities in the ontology serves as the base for a GNN, and the ontology is used to build a corpus for fine-tuning an LLM; then both the LLM and the GNN concur to the interaction prediction. Similarly, [13] also employs ontologies to produce a domain-specific corpus to be used for fine-tuning LLMs. However, in this case, the corpus is obtained by leveraging reasoning algorithms.

Instead of relying on fine tuning and inductive bias, which may be costly and difficult to implement, [14] relies on ontology-driven prompt-engineering to augment LLMs with ontological knowledge: an ontology detailing how system engineering tasks are structured is used to enhance chain-of-thought and few-shot learning strategies, improving a conversational agent in assisting system engineers with data management, requirement refining and specification clarification. The works [15] and [16] also choose to employ prompt engineering to supply ontological knowledge to an LLM, this time in the context of translating natural language questions in SPARQL. In particular, in addition to the baseline of providing the natural language question without additional information, the first work appends to the prompt the relevant classes and relations to be used, while the second work makes use of three additional prompt extensions: (i) a list of all the classes and properties of the ontology to be queried is supplied, (ii) all paths of certain length that are possible in the ontology are supplied, possibly (iii) filtered to maintain only relevant classes and properties. Both papers report a strong increase in accuracy with these prompting strategies.

While these works collectively illustrate the promise of integrating ontologies with LLMs for enhanced reasoning, accuracy, and contextual understanding, none are focused on studying the impact of supplying ontologies as structured inputs directly during the generation process itself to mitigate hallucinations. Our work fills this gap by empirically demonstrating that ontology-informed prompts improve factual reliability and constrain the output space of LLMs in knowledge-intensive applications.

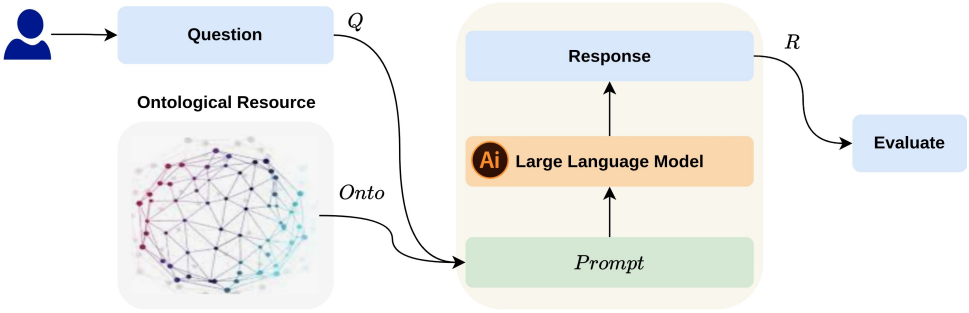

Figure 1: Overview of the proposed approach for ontology-informed LLM response generation.

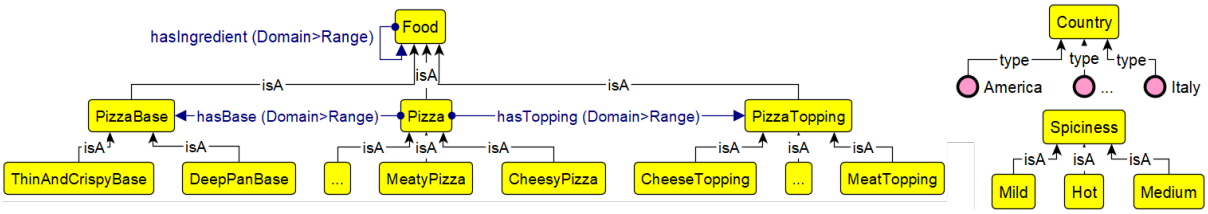

Figure 2: A representation of the main classes and properties of the Pizza Ontology, obtained using the Graffoo library for yEd (see https://essepuntato.it/graffoo/)

3. Conceptual Architecture

Figure 1 depicts a conceptual overview of the proposed method for automatic or semi-automatic LLM response generation while taking factual information from ontological resources into account. In the introduced framework, a question Q is processed alongside an ontological resource $Onto$ to construct a knowledge-enriched input. This prompt is then fed into the LLM, which generates a response R. The output may subsequently be evaluated for its relevance and correctness against gold-standard responses. In the following sections, we provide details on the ontological resources used, the prompting strategies applied, and key experimental considerations.

3.1. Ontological Resources

For our investigation, we use the popular Pizza Ontology (PO) as our main source of structured domain knowledge. Originally introduced to illustrate the capabilities of OWL-DL [17], PO has become a canonical example in the Semantic Web community. It is openly available (see https://protege.stanford.edu/ontologies/pizza/pizza.owl) and demonstrates a rich variety of OWL constructs, including domain and range restrictions, class hierarchies, qualified existential and universal quantifiers, enumerations, disjointness, cardinality constraints, and transitive properties.

PO is relatively small and well-structured, consisting of 100 classes, 8 object properties, and 800 axioms in total, of which 322 are logically more advanced. Its class hierarchy, starting from `owl:Thing`, branches into two top-level disjoint classes: `DomainThing` and `ValuePartition`. The first branch is devoted to genuine domain elements such as `Country`, `Pizza`, `PizzaTopping`, and `PizzBase`, along with various subclasses of these, both primitives (e.g. `PizzBase`, `Margherita`, etc.) and defined (e.g. `CheesyPizza`). The second branch is used to establish a list of property values, specifically the range of the `hasSpiciness` object property, which is partitioned into three classes. In addition, all primitive sister classes are disjoint. The ontology also emphasizes relationships using various object properties. For instance, `hasIngredient`, which is a transitive property subsuming `hasBase` and `hasTopping` and expresses a generic relationship between types of food.

We chose the PO for three main reasons. First, it is *well-known and widely cited*, ensuring reproducibility and familiarity within the community. Second, it models a *domain that is both formalized and intuitively understood*, overlapping with common-sense knowledge that LLMs are likely to have

Prompt (.)

You are a specialized model for answering questions using ontology-based knowledge.
Refer to the following Pizza Ontology when responding to questions:
{ontology}
Provide a clear and concise answer to the following question.
Focus strictly on pizza-related topics.
Avoid extra explanation or unrelated details.
Question: {question}
Answer:

Figure 3: Illustration of the `Prompt(.)` template used for controlled prompt engineering. The `{question}` placeholder represents the competency question (Q). Gray text corresponds to the base prompt `Prompt(Q)`, red text is added in `Prompt(Q+Onto)` to incorporate ontological context, and blue text is used in `Prompt(Q+Domain)` to emphasize domain-specific constraints.

encountered during pretraining. This makes it ideal for testing whether LLMs can align their internal representations with structured semantic input. Third, its *axiomatic richness and logical depth* enable the construction of competency questions with diverse reasoning requirements, ranging from simple lookups to complex inference tasks involving subclass hierarchies, disjointness, and cardinality constraints. Additionally, working with a logically consistent and compact ontology like PO allows us to isolate and analyze the effects of ontology-grounded prompting in a controlled and interpretable experimental setting.

3.2. Grounding LLMs with Ontological Knowledge

Given a question Q and a domain ontology $Onto$, we aim to enhance the LLM response R by contextualizing it with ontologically relevant knowledge. Three distinct methodologies are proposed:
1. Evaluating LLMs' Factual Recall. To assess the extent to which LLMs alone can answer factual questions, we employ a standard zero-shot prompting setup `Prompt(Q)` according to Figure 3. This allows us to evaluate whether the models can retrieve and utilize relevant information to generate appropriate responses to user queries Q without additional information.
2. Domain-Specific Factual Retrieval. This step evaluates the extent to which LLMs can retrieve factual information when constrained to a specific domain context, as demonstrated in the `Prompt(Q + Domain)` setup. By explicitly instructing the model to focus on a particular domain – here, it is pizzas and their ingredients – we assess how effectively it can filter out irrelevant knowledge and recall domain-specific facts. This evaluation provides insight into the LLM's ability to follow contextual boundaries and generate responses aligned with specialized knowledge.
3. Injecting Ontologies for Factual Knowledge Retrieval. In this step, we explicitly incorporate ontological knowledge into the prompt using the `Prompt(Q + Onto)`. By embedding relevant ontology axioms or excerpts directly into the `Prompt(.)`, we aim to guide the LLM toward generating responses grounded in structured, formalized knowledge. This allows for evaluating the model's capacity to utilize ontological facts for answering the Q, and to assess how well it can interpret and leverage such symbolic information when generating factual responses.

4. Evaluations

4.1. Experimental Setups

Evaluation Dataset. Since the PO does not include an official set of competency questions (CQs), we constructed a tailored set of 60 questions designed to reflect the ontology's expressiveness. These questions serve both as evaluation probes and as tools to examine how large language models handle different types of reasoning—including factual recall, subclass hierarchies, property composition, and inconsistency detection—under varying prompting strategies. We organized the questions into six

reasoning categories: individual-level facts (IND), subclass/type hierarchy (SUB), property usage (PROP), disjointness and inconsistency (DISJ), cardinality constraints (CARD), and logical/semantic composition (LOGIC). These categories represent a spectrum of ontological reasoning complexity, from simple fact retrieval to more advanced semantic inference involving multiple axioms. Since the Pizza Ontology lacks concrete individuals, we use specific named concepts such as `MargheritaPizza` to simulate instance-level reasoning in the IND category. Specifically, IND questions target properties or attributes directly associated with these named concepts. A detailed overview of the categories—including definitions and the number of questions per category—is provided in Table 1. The raw datasets are available at https://github.com/HamedBabaei/OntoTruth/blob/main/dataset/60qas.md.

Table 1
Reasoning categories used in the evaluation, along with their descriptions. The stats represent the finalized sets.

Category	Reasoning Level	Description	Total
CARD	Cardinality restrictions	Uses OWL restrictions like `exactly`, `some`, `min`	4
DISJ	Disjointness and inconsistency checking	Uses `disjointWith` and detects illegal combinations	11
IND	Individual-level fact	Uses asserted knowledge about named concepts	7
LOGIC	Logical/semantic composition (complex queries)	Involves inference across multiple dimensions or conjunctions	6
PROP	Object/Data property usage	Uses properties such as `hasTopping`, `hasBase`, etc.	9
SUB	Subclass/Type hierarchy	Uses `subClassOf` or instance type inference	14
Total number for questions after excluding flawed questions.			**51**

To evaluate consistency, each question was independently answered by two annotators. Moreover, to estimate annotator agreement, we employed two complementary methods: semantic similarity scoring [18] and the LLMs-as-Judge approach [19, 20]. While similarity-based metrics provide a general sense of alignment, they often struggle to accurately capture agreement in complex ontological contexts – particularly when responses involve subtle distinctions, long-form content, or structured knowledge that's hard to quantify via simple vector similarity. Additionally, traditional methods typically overlook the ontological structure that underpins the question-answer pairs. To address these limitations, we used the LLMs-as-Judge framework with GPT-4o [21]. This method evaluates agreement by considering the full context—including the ontology, the question, and both annotators' answers—and making a holistic judgment.

Our prompt, illustrated in Figure 4, asks the Judge LLM to (1) determine whether the answers agree or disagree, (2) explain the nature of any disagreement (e.g., differences in selected entities, relations, interpretations, or omissions), and (3) synthesize a final, unified answer [22]. Moreover, we manually checked the synthesized final answers that showed a combination of both answers by annotators or a clear version of them in cases where annotators agreed. These well-represented responses are well-formatted for the evaluation of LLMs in later steps. The LLM-as-Judge evaluation can be accessed via https://github.com/HamedBabaei/OntoTruth/blob/main/dataset/AI-Judger.json.

Using the similarity-based approach, we observed an inter-annotator agreement rate of 48%. In contrast, the LLMs-as-Judge method yielded a substantially higher agreement rate of 76%, demonstrating its effectiveness in capturing more similar answers. During this evaluation, we also identified 9 questions as fundamentally flawed or ill-formed; these were excluded from subsequent analysis. So, the final QA dataset consisted of 51 questions, which were used for further evaluations. Table 2 represents examples of question and answer pairs per reasoning level categories.

Experimental Models. We evaluated a diverse set of instruction-tuned LLMs from the LLaMA-3 [23] and Qwen2.5 [24] families (see Table 3), grouped into three size categories: small, medium, and large. The small category includes `Llama-3.2-1B-Instruct`[2] and `Qwen2.5-0.5B-Instruct`[3], representing models with fewer than 1B parameters, suitable for lightweight reasoning tasks. The medium category consists of `Llama-3.1-8B-Instruct`[4] and `Qwen2.5-7B-Instruct`[5], offering a

[2]https://huggingface.co/meta-llama/Llama-3.2-1B-Instruct
[3]https://huggingface.co/Qwen/Qwen2.5-0.5B-Instruct
[4]https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct
[5]https://huggingface.co/Qwen/Qwen2.5-7B-Instruct

Table 2
Examples of questions and answers per category from the evaluation dataset.

Category	Question	Answer
CARD	Which pizzas are built using a ThinAndCrispyBase?	Real Italian Pizza, Napoletana, Veneziana
DISJ	What toppings are considered disjoint from MeatTopping?	VegetableTopping, CheeseTopping, FishTopping, HerbSpiceTopping, FruitTopping, NutTopping, SauceTopping
IND	What are the common toppings of a Margherita Pizza?	Mozzarella and tomato
LOGIC	What spicy meat topping is commonly found on American-style pizzas?	Peperoni Sausage
PROP	Which pizzas contain both OnionTopping and PepperTopping?	Pollo Ad Astra, Sloppy Giuseppe, Cajun
SUB	Is a QuattroFormaggiPizza classified as a CheesePizza?	Yes

LLMs-as-Judge for Annotator Agreements

I have two annotators who answered the same question based on an ontology.
Compare their answers and judge whether they agree or disagree.
If they disagree, explain the nature of the disagreement (e.g., different entities, relations, interpretations, or missing concepts).
Next, provide a single answer based on two annotators. This response should be straightforward with no extra explanation.

<question>
question
</question>

<ontology>
ontology
</ontology>

<annotator-1-answer>
annotator1
</annotator-1-answer>

<annotator-2-answer>
annotator2
</annotator-2-answer>

Return your output in the following format:

{'agreement': 'agree', 'rationale': '...', 'answer': '...'}

Figure 4: LLMs-as-Judge for annotator agreements.

Table 3
LLM Families.

LLM	Context Lenght	Source
LLaMA-3 Family	131K (≈100K words)	Hugging Face
Qwen2.5 Family	32K (≈25K words)	Hugging Face

trade-off between performance and computational efficiency. Finally, the large category includes `Llama-3.3-70B-Instruct`[6] and `Qwen2.5-72B-Instruct`[7], representing high-capacity models expected to perform best on complex ontological reasoning tasks. This categorization enables us to systematically investigate how model scale affects performance across various reasoning categories. In addition, we are experimenting with different materializations of inferred consequences across ontology

[6] https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct
[7] https://huggingface.co/Qwen/Qwen2.5-72B-Instruct

Table 4
Mean cosine similarity (in %) between gold and generated responses, using different LLMs across model sizes (Small, Medium, Large). The **Prompt(Q)** column represents performance without using ontology, while **Prompt(Q+Domain)** indicates a domain-specific specification added to the prompt. The remaining columns show results using different materialized ontology formats within **Prompt(Q+Onto)**: RDF/XML, Manchester OWL Syntax, OWL Functional Syntax, and Turtle. Bolded values indicate the best performance within each row.

Category	LLM	Prompt(Q)	Prompt(Q+Domain)	Prompt(Q+Onto)			
				RDF	Manchester	Functional	Turtle
Small	*Qwen2.5-0.5B*	39.7	**40.5**	34.7	12.7	29.3	28.6
	Llama-3.2-1B	39.5	40.6	44.3	**44.8**	44.3	43.4
Medium	Qwen2.5-7B	43.8	45.1	**57.1**	52.9	51.4	53.3
	Llama-3.1-8B	44.2	44.9	**54.2**	50.1	52.2	52.1
Large	*Qwen2.5-72B*	45.7	45.4	**60.3**	56.9	50.4	57.2
	Llama-3.3-70B	46.4	45.7	**55.2**	48.1	39.7	46.8

formats to understand how the explicit presence of derived knowledge affects LLM reasoning. The formats used include RDF/XML, Manchester OWL Syntax, OWL Functional Syntax, and Turtle.

Evaluation Metrics. For automatic evaluation of generated responses, we computed the cosine similarity between each LLM-generated answer and the corresponding gold answer using embeddings from OpenAI's text embedding model (specifically `text-embedding-3-large`[8]). This approach provides a lightweight semantic similarity metric to estimate alignment between model outputs and human-annotated references.

4.2. Results

RQ1. To what extent can domain-specific ontological knowledge reduce hallucinations and factual errors in LLM outputs? According to Table 4, our results demonstrate that incorporating domain-specific ontological knowledge substantially reduces hallucinations and factual inaccuracies in LLM-generated answers. As shown in Table 4, performance with only the question prompt (`Prompt(Q)`) yields relatively low alignment with gold answers across all model sizes—averaging between ≈39–46%—indicating a considerable risk of hallucinations due to a lack of grounding.

Improvements obtained by introducing a lightweight domain-specific hint (`Prompt(Q + Domain)`) are marginal at best (not more than 1.3% gain), suggesting that merely indicating the domain context is insufficient to significantly reduce factual errors. The most notable improvement is observed when using `Prompt(Q + Onto)`, where explicit ontological content is hard-coded into the prompt. For example, larger LLMs like Qwen2.5-72B and Llama-3.3-70B achieve 60.3% and 55.2% alignment, respectively, under RDF materialization—an increase of over 10 percentage points compared to domain-only `Prompt(Q + Domain)` or question-only `Prompt(Q)` prompts. Similar trends are observed across medium and small models, confirming that access to formal ontology structures consistently boosts performance. This highlights that LLMs are indeed capable of retrieving information from materialized ontologies and using it in their generated answers.

Further, Table 5 reveals that the benefits of ontological grounding are particularly pronounced in reasoning-heavy categories such as *IND* (individual-level facts), *SUB* (subclass hierarchy), and *PROP* (property usage), where structured definitions and relationships help models infer correct answers and avoid hallucinated entities or unsupported claims. For instance, in the IND category, the accuracy of Qwen2.5-7B rises dramatically from 51.2% to 82.3% with ontology grounding.

However, the results for small-scale LLMs show an opposite trend in certain categories, where grounding sometimes reduces performance. We attribute this drop to the limited capacity of smaller models to effectively parse and exploit structured ontology materializations, which can increase cognitive

[8] https://openai.com/index/new-embedding-models-and-api-updates/

Table 5
Category-wise cosine similarity performance (%) of each LLM across two evaluation settings: **Prompt(Q)** denoted as a (P) and RDF-materialized ontology using **Prompt(Q+Onto)** denoted as a P'. Results are grouped by reasoning category (CARD, DISJ, IND, LOGIC, PROP, SUB) and model size (Small, Medium, Large).

Category	LLM	CARD		DISJ		IND		LOGIC		PROP		SUB	
		P	P'	P	P'	P	P'	P	P'	P	P'	P	P'
Small	*Qwen2.5-0.5B*	43.9	49.2	43.5	32.0	47.6	35.8	32.1	26.8	32.4	30.3	39.8	38.5
	Llama-3.2-1B	48.1	42.8	44.0	52.8	47.2	47.4	30.7	32.1	28.8	37.8	40.5	46.1
Medium	*Qwen2.5-7B*	34.9	41.7	43.2	51.5	51.2	82.3	41.1	56.2	31.7	40.9	52.3	64.4
	Llama-3.1-8B	31.5	53.1	41.3	50.8	49.9	74.2	38.3	32.8	35.6	49.2	55.5	59.6
Large	*Qwen2.5-72B*	46.7	46.9	51.5	57.9	54.4	78.2	33.9	45.4	37.6	65.1	46.9	60.4
	Llama-3.3-70B	33.8	55.8	44.7	50.6	58.7	82.7	46.2	36.0	37.2	57.3	51.3	51.8

load instead of providing helpful constraints. In these cases, the additional symbolic structure may overwhelm the model's limited reasoning ability, leading to noisier outputs.

In summary, domain ontologies act as strong factual anchors for LLMs. By injecting structured, semantically rich knowledge directly into prompts, hallucinations and factual drift are significantly reduced—especially in tasks that demand ontological reasoning. This highlights the value of knowledge-informed prompting for reliable AI-assisted reasoning in complex domains.

RQ2. What are the practical challenges and limitations when providing ontological sources as part of an LLM prompt? Our experiments and framework design reveal the following key issues:

- **Complexity and Format of Ontologies.** Ontologies can be encoded in multiple syntactic materializations of consequences—RDF/XML, Manchester OWL, OWL Functional Syntax, Turtle—each with different levels of human readability and parsing complexity. As shown in Table 4, model performance varies substantially across these materializations, even when the underlying knowledge remains the same. For example, a large LLM like Llama-3.3-70B performs best with RDF, but shows degraded performance with Functional Syntax (e.g., 55.2% vs. 39.7%), likely due to increased syntactic overhead. This illustrates a key limitation: "*not all ontology encodings are equally digestible to LLMs, and some may obscure rather than clarify semantic relationships*".
- **Context Length.** Injecting ontological knowledge into prompts significantly increases context length (See Table 3, this study's LLMs are capable of supporting 32K to 131K input limits). Large ontologies can contain hundreds of axioms, many irrelevant to the current query. While LLMs like GPT-4o can handle long contexts, smaller models struggle with salience filtering—i.e., identifying which parts of the ontology are relevant to a given question. This leads to noisy or hallucinated outputs due to *cognitive overload of key signals*.
- **Logical Axiom Processing.** Ontologies often include constructs like `disjointWith`, `someValuesFrom`, and `cardinality` constraints, which require non-trivial logical reasoning. As seen in Table 5, categories like CARD (cardinality) and LOGIC (complex reasoning) exhibit lower and more variable averaged similarity scores even with ontology prompts. For example, performance in LOGIC questions for Llama-3.3-70B drops from 46.2% (Prompt only) to 36.0% (with ontology), indicating that the presence of complex axioms can confuse models rather than help them, unless carefully abstracted.
- **Beyond Hard-Coded Knowledge.** Unlike symbolic reasoners, LLMs lack built-in mechanisms to dynamically query ontologies or perform deductive closure. Instead, they rely on pattern recognition and latent knowledge, rather than formal, rule-based inference. This limits their ability to validate claims that require multi-step deductions, indirect class/property chains, or deeper reasoning over ontological structures. The static nature of prompt-based injection—where entire ontologies are hard-coded into the input—further restricts the model's ability to focus on relevant portions of the knowledge base. This challenge underscores the promise of RAG enabled by ontologies, where relevant triples or schema fragments can be selectively retrieved in real time

to guide LLM responses. Such systems could support goal-directed navigation over ontologies, reducing context overload and enabling more accurate, semantically grounded generation without requiring full ontology injection into every prompt.

In summary, while ontological resources provide valuable structured knowledge for LLM validation, practical challenges related to ontology complexity, context limitations, logical reasoning, and the lack of dynamic querying capabilities currently hinder their effective integration. Addressing these limitations—particularly through approaches like RAG that enable selective, real-time ontology access—holds promise for more robust and semantically grounded validation of LLM-generated claims.

5. Conclusions and Future Directions

In this paper, we presented an investigation for integrating ontological knowledge into LLM workflows to improve factual accuracy. Using the pizza ontology as a case study, we showed that ontology-informed prompts significantly reduce hallucinations. This confirms the importance of injecting domain-specific ontological knowledge as a strong factual anchor for LLM generation. While effective for small, curated ontologies, scaling this approach remains challenging due to token limits and the lack of formal reasoning in LLMs. To overcome these limitations, we highlight the need for hybrid architectures that combine LLMs with automated reasoners, and propose reinforcement learning with reasoning feedback as a promising future direction. In future work, we plan to systematically evaluate the use of summarized ontologies for prompt enrichment and assess their trade-offs in terms of accuracy improvements and computational efficiency.

Acknowledgments

Hamed Babaei Giglou was supported by the Federal Ministry of Research, Technology and Space, Germany (BMFTR) through the SCINEXT project (Grant ID: 01lS22070). Sebastian Rudolph was supported by the BMFTR and DAAD in project 57616814 “School of Embedded Composite Artificial Intelligence” (SECAI).

Declaration on Generative AI

In preparing this manuscript, generative AI tools – specifically ChatGPT – were used solely for: grammar checking, spelling check, and the readability of some sentences. All suggested changes were carefully reviewed and adapted by the authors to ensure accuracy and appropriateness. The scientific content, research design, analysis, and conclusions were developed and verified exclusively by the authors without AI involvement. The use of ChatGPT was limited to enhancing the presentation of the work.

References

[1] R. Alharbi, U. Ahmed, D. Dobriy, W. Łajewska, L. Menotti, M. J. Saeedizade, M. Dumontier, Exploring the role of generative ai in constructing knowledge graphs for drug indications with medical context, Proceedings http://ceur-ws. org ISSN 1613 (2023) 0073.

[2] J. E. Kim, C. Henson, K. Huang, T. A. Tran, W.-Y. Lin, Accelerating road sign ground truth construction with knowledge graph and machine learning, 2020. URL: https://arxiv.org/abs/2012.02672. arXiv:2012.02672.

[3] H. Li, G. Appleby, K. Alperin, S. R. Gomez, A. Suh, Mitigating LLM hallucinations with knowledge graphs: A case study, 2025. URL: https://arxiv.org/abs/2504.12422. arXiv:2504.12422.

[4] R. I. M. Vsevolodovna, M. Monti, Enhancing large language models through neuro-symbolic integration and ontological reasoning, 2025. URL: https://arxiv.org/abs/2504.07640. arXiv:2504.07640.

[5] D. Allemang, J. Sequeda, Increasing the LLM accuracy for question answering: Ontologies to the rescue!, 2024. URL: https://arxiv.org/abs/2405.11706. arXiv:2405.11706.
[6] M. Monti, O. Kutz, G. Righetti, N. Troquard, Improving the accuracy of black-box language models with ontologies: a preliminary roadmap, in: Proceedings of the Joint Ontology Workshops 2024, 2024.
[7] T. Baldazzi, D. Benedetto, L. Bellomarini, E. Sallinger, A. Vlad, Softening ontological reasoning with large language models, in: 8th International Joint Conference on Rules and Reasoning co-located with 20th Reasoning Web Summer School (RW 2024) and 16th DecisionCAMP 2024 as part of Declarative AI 2024, 2024. URL: https://ceur-ws.org/Vol-3816/paper33.pdf.
[8] M. U. Din, J. Rosell, W. Akram, I. Zaplana, M. A. Roa, L. Seneviratne, I. Hussain, Ontology-driven prompt tuning for LLM-based task and motion planning, 2024. URL: https://arxiv.org/abs/2412.07493. arXiv:2412.07493.
[9] E. Motta, F. Osborne, M. M. L. Pulici, A. Salatino, I. Naja, Capturing the Viewpoint Dynamics in the News Domain, Springer Nature Switzerland, 2024, p. 18–34. URL: http://dx.doi.org/10.1007/978-3-031-77792-9_2. doi:10.1007/978-3-031-77792-9_2.
[10] H. T. Mai, C. X. Chu, H. Paulheim, Do LLMs Really Adapt to Domains? An Ontology Learning Perspective, Springer Nature Switzerland, 2024, p. 126–143. URL: http://dx.doi.org/10.1007/978-3-031-77844-5_7. doi:10.1007/978-3-031-77844-5_7.
[11] F. Ronzano, J. Nanavati, Towards ontology-enhanced representation learning for large language models, 2024. URL: https://arxiv.org/abs/2405.20527. arXiv:2405.20527.
[12] T. V. Ivanisenko, P. S. Demenkov, V. A. Ivanisenko, An accurate and efficient approach to knowledge extraction from scientific publications using structured ontology models, graph neural networks, and large language models, International Journal of Molecular Sciences 25 (2024). URL: https://www.mdpi.com/1422-0067/25/21/11811. doi:10.3390/ijms252111811.
[13] T. Baldazzi, L. Bellomarini, S. Ceri, A. Colombo, A. Gentili, E. Sallinger, Fine-tuning large enterprise language models via ontological reasoning, in: A. Fensel, A. Ozaki, D. Roman, A. Soylu (Eds.), Rules and Reasoning, Springer Nature Switzerland, Cham, 2023, pp. 86–94.
[14] J. Gauthier, E. Jenn, R. Conejo, Ontology-driven LLM assistance for task-oriented systems engineering, in: Proceedings of the 13th International Conference on Model-Based Software and Systems Engineering - MBSE-AI Integration, INSTICC, SciTePress, 2025, pp. 383–394. doi:10.5220/0013441100003896.
[15] A. Gashkov, A. Perevalov, M. Eltsova, A. Both, SPARQL query generation with LLMs: Measuring the impact of training data memorization and knowledge injection, 2025. URL: https://arxiv.org/abs/2507.13859. arXiv:2507.13859.
[16] M. Mountantonakis, Y. Tzitzikas, Generating SPARQL queries over CIDOC-CRM using a two-stage ontology path patterns method in LLM prompts, J. Comput. Cult. Herit. 18 (2025). URL: https://doi.org/10.1145/3708326. doi:10.1145/3708326.
[17] A. Rector, N. Drummond, M. Horridge, J. Rogers, H. Knublauch, R. Stevens, H. Wang, C. Wroe, OWL pizzas: Practical experience of teaching OWL-DL: Common errors & common patterns, volume 3257, 2004, pp. 63–81. doi:10.1007/978-3-540-30202-5_5.
[18] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2019. URL: https://arxiv.org/abs/1908.10084.
[19] H. Li, Q. Dong, J. Chen, H. Su, Y. Zhou, Q. Ai, Z. Ye, Y. Liu, LLMs-as-judges: a comprehensive survey on LLM-based evaluation methods, arXiv preprint arXiv:2412.05579 (2024).
[20] J. D'Souza, H. B. Giglou, Q. Münch, Yescieval: Robust LLM-as-a-judge for scientific question answering, arXiv preprint arXiv:2505.14279 (2025).
[21] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, et al., Gpt-4o system card, arXiv preprint arXiv:2410.21276 (2024).
[22] H. Babaei Giglou, J. D'Souza, S. Auer, Llms4synthesis: Leveraging large language models for scientific synthesis, in: Proceedings of the 24th ACM/IEEE Joint Conference on Digital Libraries, 2024, pp. 1–12.

[23] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan, et al., The llama 3 herd of models, arXiv e-prints (2024) arXiv–2407.

[24] A. Yang, B. Yu, C. Li, D. Liu, F. Huang, H. Huang, J. Jiang, J. Tu, J. Zhang, J. Zhou, et al., Qwen2. 5-1m technical report, arXiv preprint arXiv:2501.15383 (2025).