Conceptual Diagnostics for Knowledge Graphs and Large Language Models

Rosario Uceda-Sosa, Maria Chang,
Karthikeyan Natesan Ramamurthy, Moninder Singh
IBM Research
{rosariou, knatesa, moninder}@us.ibm.com, maria.chang@ibm.com

Abstract

Industrial applications pose heightened requirements for consistency and reliability of large language models (LLMs). While LLMs are being tested with increasingly complex reasoning tasks, we argue that much can be learned via diagnostic tools that probe a fundamentally basic type of reasoning: conceptual consistency, e.g., a rule applying to "all surgeons" must also apply to "cardiac surgeons" since a cardiac surgeon is a type of surgeon. In this emerging industry track submission, we propose a method that takes concept hierarchies from a knowledge graph (KG) and automatically generates benchmarks that test conceptual consistency in LLMs. We develop a multi-domain benchmark that reveals rates of conceptual inconsistencies in several state of the art LLMs. Additionally, we use measured levels of inconsistency and disagreement in LLMs to find potentially problematic subgraphs in the reference KG. As such, it offers a scalable complement to symbolic curation, maintenance, and refinement of knowledge graphs, which is a critical activity in KG-based industrial applications.

1 Introduction

Large Language Models (LLMs), despite their tremendous success on traditional benchmarks, often commit errors that limit their application in real-world industrial settings (Haltaufderheide and Ranisch, 2024; Zhang et al., 2025; Dahl et al., 2024). Reliability and consistency of LLMs (Xu et al., 2024; Ji et al., 2023) are key issues that undermine performance and trust. Developing diagnostic tools that can measure the reliability of LLMs in a way that is principled, scalable, and application-domain-focused, is very difficult. Yet, it is critical for high-stakes industrial domains like healthcare, law, or manufacturing, where unpredictable behavior can have serious consequences.

Much attention has been given to LLM abilities on complex tasks that are challenging for even the most highly trained humans (Jaech et al., 2024). Although very impressive, we argue that diagnostic tools can be built via a much more basic type of reasoning: *conceptual consistency*. Conceptual consistency is the ability to reliably produce equivalent answers to semantically equivalent queries about a conceptual hierarchy. It is basic because it concerns the fundamental categorization and property inheritance of concepts. For example, a rule applying to "all surgeons" must naturally extend to "cardiac surgeons" since a cardiac surgeon is a type of surgeon - this is a basic generalization that hinges on a stable conceptual framework. Furthermore, when an LLM is asked about the conceptual hierarchy of surgeons, it should not change its answer when it is asked in a slightly different but semantically equivalent way. This is especially important in real-world applications, where organizations need to verify that the models are aligned with domain-specific knowledge bases, such as product

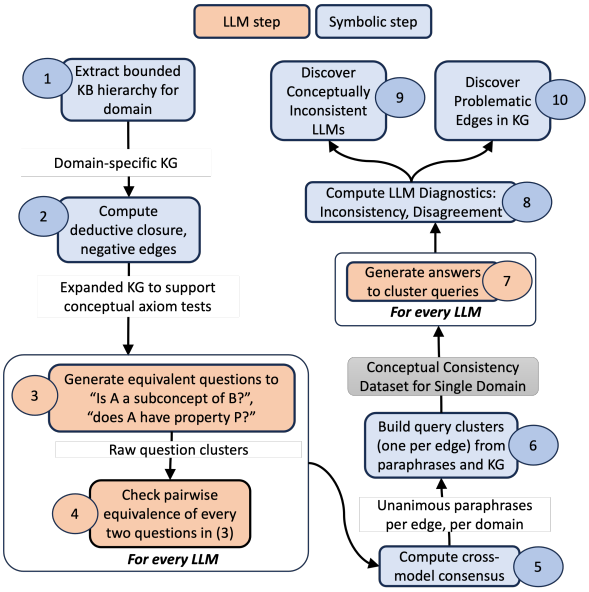

Figure 1: Proposed automated conceptual diagnostics pipeline for a single dataset.

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track), pages 531–540
July 28-30, 2025

catalogs, medical specialists taxonomies, scientific corpora, and so on.

Knowledge graphs (KGs), on the other hand, are conceptually consistent by design, but have their own set of issues. One of the biggest challenges in using them in industrial applications is maintaining them to ensure their knowledge is factual, up to date, and as complete as necessary for its downstream task. With very large KGs, curating and repairing knowledge can be a substantial obstacle.

We propose a method to automatically generate, with a domain-agnostic process, domain-specific benchmarks that assess the conceptual consistency of LLMs. This domain-agnostic process facilitates generalization, while the creation of domain-specific benchmarks is suited to many industrial applications. The same process can be used to generate benchmarks for finance products, home appliances, medical specialties, and so on (Table 1). Furthermore, we show that analytics from our benchmark can be used to discover areas of the KG that are problematic and need human attention. We illustrate our method on 4 well-established LLM families and 8 domains from the Wikidata KG. These experiments provide empirical support for our method and a pathway to its deployment.

This work has the following contributions:

1. We introduce a domain-agnostic method for creating benchmark datasets that test conceptual self-consistency in LLMs.

2. We release a new benchmark dataset to test conceptual self-consistency in LLMs that consists of over 6,000 deducible edges and 30,000 LLM queries across 8 distinct domains extracted as a KG from Wikidata[1].

3. We show that in addition to revealing inconsistencies in state of the art LLMs, these benchmarks can be used to identify representational errors and problematic subgraphs in the source KG.

Figure 1 shows the methodological contributions of our work, discussed in detail in Sections 3 and 5.

The rest of this paper is organized as follows. We begin with preliminaries regarding conceptual hierarchies (Section 2) followed by our core methodology (Section 3). We present our findings across several domains and LLMs (Section 4) and propose a feedback mechanism for discovering problems with the source KG (Section 5). We conclude with directions for future research (Section 7) and limitations (Section 8).

[1] https://huggingface.co/datasets/ibm-research/knowledge_consistency_of_LLMs

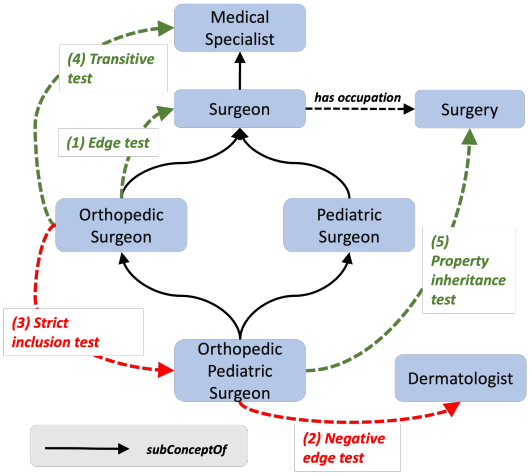

Figure 2: Concept axiom tests (dotted edges numbered 1-5) shown on an example concept hierarchy (solid lines) of medical specialist.

2 Conceptualization properties

Webster defines a *concept* as "an abstract or generic idea generalized from particular instances." Similarly, a *type* is "a particular kind, class, or group". Either of these definitions refer to a set of instances that share similar properties and can be organized into a generalization hierarchy (Brachman and Levesque, 2004). Operationally, we define a concept C as a set of instances. For example, the concept “land vehicle” represents a broad category that includes instances of cars, trucks, motorcycles, etc. and they all have a propulsion system, a steering system, the ability to transport people or goods and so on.

The *subconcept* relation (also known as an “is-a” or ”subclass of” relationship or taxonomy) is a hierarchical relationship where a more specific concept (the subconcept) inherits the properties of a broader, more general concept (the parent concept), while the parent concept inherits the instances of its subconcepts. An illustration of subconcept relations in the medical specialities domain is shown in Figure 2.

Given a concept hierarchy with subconcept relations, a set of concept axioms may be used to compute the *deductive closure* of the graph, which is the full set of edges that can be inferred from the set of explicit edges. The following axioms are

used to compute the deductive closure of the conceptual hierarchy: (1) edge reflexivity/identicality, which simply asserts the existence of a known edge, (2) negative edge, in which the absence of an edge implies its negation, (3) strict inclusion, which prevents subconcept cycles in the hierarchy, (4) transitivity, which enables transitive inference of subconcept relations, and (5) property inheritance, which asserts that if a property exists for a given concept, then it also exists for all corresponding subconcepts. Property inheritance is especially powerful, as is underpins the utility and coherence of structured concept hierarchies. The hierarchy in Figure 2 shows how these axioms indicate that some edges are part of the deductive closure (green edges 1,4,5), while other edges contradict it (red edges 2,3). Following (Uceda-Sosa et al., 2024), we evaluate the conceptual consistency of LLMs with respect to the most fundamental elements of the conceptual hierarchy: the basic subconcept relations and a single property. We use tests that are based on the concept axioms described above.

3 Building benchmarks to test conceptual consistency

We aim to automatically generate datasets that evaluate the conceptual consistency of large language models (LLMs) with respect to a concept hierarchy. Due to the proprietary and sensitive nature of most customer data, we adopt the Wikidata concept hierarchy as an open and structured knowledge base (Vrandečić, 2012; Erxleben et al., 2014; Vrandečić and Krötzsch, 2014; Voß, 2016) whose contents are widely available.

We focus on eight distinct domain-specific datasets encompassing concepts at varying levels of abstraction and ontological persistence (Borgo et al., 2023), spanning from concrete entities such as software products, financial services, and household appliances, to more abstract categories like music genres, academic disciplines, and event types such as natural disasters (Table 1). While the top-level concepts and properties are manually selected, the associated subgraphs are retrieved automatically using the Wikidata public SPARQL endpoint.[2]

The pipeline to create these datasets is depicted in Figure 1. Steps in blue are symbolic in nature, while the orange steps are executed by the LLMs. We start by extracting a concept hierarchy based on a top concept plus one property and a curated set of 10–20 seed leaf concepts per domain. We select these seed concepts for expediency of results, since some of these hierarchies may have tens of thousands of leaves, but it is by no means a compulsory step. Practitioners may decide to automatically process all possible leaves in a hierarchy, provided they have the computational power.

Domain	Predicate	C	$\frac{Q}{C}$
Academic Disciplines	used for	443	4.20
Dishes	has ingredient	1220	5.15
Finance Products	used for	725	4.57
Home Appliances	used for	421	5.67
Medical Occupations	has occupation	740	4.94
Music Genres	practiced by	1990	6.09
Natural Disasters	has cause	357	4.52
Software	studied in	249	4.49

Table 1: Sample domains in benchmark; number of clusters denoted by C; number of questions per cluster denoted by $\frac{Q}{C}$

The top concept and leaf nodes create a bounded, domain-specific KG (step 1). While it is feasible to automatically process all -or randomly selected- leaf concepts across the full hierarchy, yielding significantly larger domain-specific KGs, we found that even this modest sampling reveals substantial inconsistencies and allows us to easily bypass esoteric concepts and less informative (e.g. bookkeeping) edges. Next, we compute the deductive closure of the hierarchy and arbitrary negative edges to test (step 2). The resulting KG consists of a set of domain-specific concepts, the subconcept-of relationships between them, one property (e.g. 'has occupation' in Figure 2), and additional edges that enable axiom tests.

Our goal isn't to check whether LLMs perfectly match the domain-specific knowledge graph (KG), but whether they are consistent with their own internal understanding of the conceptual hierarchy. To test this, we rely on the models themselves to generate semantically equivalent paraphrases of each edge (either physical or virtual) in the hierarchy (step 3). When multiple models agree on these paraphrases within a domain, we then test them further by inserting real examples from the KG (step 4). Finally, we check again across models to make sure they all still treat the paraphrased queries as having the same meaning (step 5).

It is worth noting that not all paraphrases are equivalent across domains, just like not all queries are relevant to all domains. For example, "Is every

[2] https://query.wikidata.org

↓ LLM responses	pred(A,B)	¬pred(A,B)
All YES	CA	CD-FP
All NO	CD-FN	CA
Mixed YES,NO	Inconsistent	Inconsistent

Table 2: Breakdown of possible LLM behaviors in our consistency benchmark: consistent agreement (CA), consistent disagreement (CD) with false positive (-FP) and false negative (-FN) variants. pred(A, B) indicates that entity A is related to entity B through a relationship (predicate). ¬pred(A,B) is the negation of it.

X a Y?" does not make sense in academic disciplines. You can't ask "Is every algebra a mathematics?" However, in medical specialties, "Is every orthopedic surgeon a surgeon?" makes sense This is why the steps 3, 4 and 5 above need to be domain specific.

Next, we build the dataset, creating *query clusters*, sets of questions designed to evaluate edges within the concept hierarchy (step 6)—whether explicitly stated, inferred through deductive closure, or deliberately constructed as a non-existent (i.e., false) edge, as illustrated in Figure 2. Despite their differing origins, all clusters share the property that their constituent questions are expected to elicit a uniform binary response: either all 'yes' (denoting a positive edge cluster, shown in green) or all 'no' (denoting a negative edge cluster, shown in red). For this reason, we refer to them collectively as binary agreement (BA) clusters.

The majority of BA clusters in our dataset test individual edges using sets of four semantically equivalent paraphrased questions. These canonical clusters form the basis for assessing local conceptual consistency. A subset of the positive edge clusters, however, evaluate virtual relations, such as those implied by transitivity or property inheritance, present only in the deductive closure of the graph. These cases are represented by higher-order conceptualization tests, with an antecedent and a consequent. For example, in the case of transitivity we may have in the antecedent the edges 'A subconcept of B' and 'B subconcept of C' and, in the consequent 'A subconcept of C'. The corresponding BA clusters for these axioms involve multiple sets of semantically equivalent queries, each testing both antecedents and consequents. While not all questions in these extended clusters are paraphrases of each other, the expectation of binary agreement still holds: the model should answer consistently across all questions within a cluster.

Empirical evidence supporting the validity of our approach is reflected in the high agreement rate among models: across all tested domains (see Section 4 below), LLMs provide consistent and correct answers to the generated queries in approximately 90% of cases, underscoring the effectiveness of our method in probing conceptual consistency.

4 Evaluation

Irrespective of the specific paraphrasing, all binary agreement (BA) clusters, by construction, elicit a uniform binary response, either 'yes' or 'no'. If the LLM answers the entire cluster uniformly and with an answer that is consistent with the KG, then the cluster is marked **consistent agreement**. Conversely, if the model answers the entire cluster uniformly but contradicts the truth label derived from the knowledge base (e.g., uniformly answering yes to a cluster that corresponds to an edge that doesn't exist in the KG), we classify the cluster as having a **consistent disagreement**. Only when the LLM responds to semantically equivalent questions with a mixture of yes and no responses is the cluster marked conceptually inconsistent. Table 2 shows these conditions as a truth table.

We have evaluated the benchmarks described above using four model families: DeepSeek, Llama, Granite and Mistral (Figure 3).

As we see in Figure 3, LLMs reason inconsistently on approximately 10% of clusters, regardless of model size or version. It is worth noting that all LLMs tested show some level of inconsistency, although some domains, like 'software', seem to be more reliable than others. In particular, we see that 'music genres' seems to be an outlier in terms of consistency.

Within the set of consistent clusters, consistent disagreements occur in approximately 2% of all evaluated clusters across all LLMs. The highest rate of consistent disagreement for any given cluster-LLM combination is less than 6% (see Appendix for detailed statistics). Despite their relative rarity, consistent disagreement clusters appear across all tested LLMs and domains, with the sole exception of DeepSeek-V2 in the software domain.

An additional layer of insight emerges when analyzing the polarity of these disagreements. We estimate the proportion of consistent disagreement clusters in which the LLM asserts the existence of edges that are absent in the source KG (CD-FP in Table 2). These can be thought of as consistent

model	academic disciplines	dishes	finance	home appliances	medical specialties	music genres	natural disasters	software	AVG
DeepSeek-r1	12.07	8.94	14.9	10.48	11.35	18.73	11.2	8.84	12.06
DeepSeek-V2_5	6.15	5.75	10.9	5.9	8.65	14.47	12.89	2.81	8.44
DeepSeek-V3	11.16	8.12	11.45	12	13.65	22.1	15.13	8.84	12.81
granite-3_0-8b-instruct	7.74	10.58	9.52	9.52	13.38	20.37	12.32	8.43	11.48
granite-3_1-8b-instruct	8.88	5.57	10.9	7.05	8.38	20.83	11.2	6.43	9.91
granite-3_2-8b-instruct	8.66	5.66	10.62	6.86	8.24	20.55	11.2	6.43	9.78
llama-3-1-70b-instruct	10.93	5.93	15.93	10.1	10.07	18.33	15.13	5.42	11.48
llama-3-3-70b-instruct	11.39	6.11	14.14	9.71	8.24	16.95	11.48	5.22	10.41
mistral-large-instruct-2407	10.71	7.3	11.17	10.67	10.41	19.85	11.48	8.43	11.25
mixtral-8x22B-instruct-v0_1	8.66	7.48	11.59	9.33	10.14	18.82	9.24	8.43	10.46
mixtral-8x7B-instruct-v0_1	8.66	7.94	11.03	7.05	12.03	19.66	12.61	9.64	11.08
AVG	9.68	7.23	12.09	8.98	10.55	19.23	12.32	7.15	10.90

Figure 3: Percentages of inconsistent clusters by model and domain.

hallucinations with respect to the KG. These account for approximately 15% of an already small subset of clusters (see Appendix for details). This means that the dominant trend in LLM disagreement involves false negatives (CD-FN in Table 2), where the model systematically denies edges that are present in the KG.

Finally, we observe that neither architectural scale nor newer model versions significantly mitigate the observed inconsistencies. This suggests that such structural inconsistencies are not merely artifacts of model size or versioning, but are instead deeply rooted in the underlying training data and inductive biases of current LLM architectures. Addressing these limitations may require architectural innovations or fundamentally new approaches to knowledge representation and reasoning in LLMs.

5 Identifying problematic subgraphs

As noted above, conceptual consistency does not depend on uniform agreement with the reference KG. Although community curated KGs such as Wikidata are very rich approximations of world knowledge, we cannot treat them as definitive ground truth. Indeed, curating, validating, and maintaining KGs is a significant challenge for industrial applications that use them. In this section we show that LLM consensus can be leveraged to identify and potentially resolve ambiguous or conflicting edges in the underlying KG.

We consider two types of evidence that parts of the KG are subjective, incorrect, or otherwise problematic from a knowledge modeling perspective: occurrence of consistent cluster disagreement and rate of edge disagreement. If a particular domain was factually incorrect, we would expect the clusters for that domain to have a high rate of consistent disagreement across several LLMs. However, as noted in Section 4, this only occurs approximately 2% of the time across all domains and LLMs, which is not a strong signal of incorrectness at the domain level. To get a more detailed picture, we measure the rate of edge disagreement, which is the proportion of queries on which the LLM disagrees with the KG, irrespective of the consistency of the LLM reasoning.

This approach proves particularly insightful in the case of the music genres domain, which consistently emerges as an outlier across all evaluated models. As illustrated in Figure 4, the distribution of disagreements exhibits a long tail: the top 100 edges on which LLMs most often disagree account for 48.8% of all disagreements across LLMs. Notably, the majority of disagreements occur around three semantically dense regions of the subgraph: English folk and country music, Jamgrass, and Christmas-themed genres such as carols and hymns. The Wikidata hierarchies in this domain are deep, with many detailed categorizations that may not be standard across knowledge bases. There may also be some disagreement in the meaning of some terms, as in 'country music', which can be equated with 'folk music' or can be understood as a more specific genre specific to North America (US and Canada) by some Wikidata contributors. This points to the challenges of modeling complex domains, particularly those characterized by soft taxonomies, federated authorship or overlapping conceptual boundaries. In such cases, even small inconsistencies or modeling decisions can lead to cascading effects in inference and reasoning. Leveraging the probabilistic consensus of LLMs may offer a scalable complement to symbolic curation, suggesting a novel avenue for semi-automated KG refinement, and helping to surface latent ambiguities to improve KG robustness over time.

edge	% Disagreement
English_country_music--subconcept--English_folk_music	1.57
Irish_folk_music--subconcept--British_folk_music*	1.55
Christmas_carol--subconcept--music_genre	1.25
Christmas_hymn--subconcept--music_genre	1.13
English_country_music--subconcept--Celtic_folk_music	1.03
English_folk_music--subconcept--Celtic_folk_music	1.03
English_country_music--subconcept--British_folk_music	1.02
British_folk_music--subconcept--Celtic_folk_music	0.99
jamgrass--subconcept--traditional_country *	0.97
progressive_bluegrass--subconcept--traditional_country	0.97
⋮	

First 100 edges = 48 nodes = 48.8% of disagreement

Figure 4: Frequency of edge disagreement across LLMs. *Examples of edges that are subjective and possibly incorrect in the KG.

6 Related work

The idea that LLMs implicitly encode relational knowledge, traditionally stored in symbolic knowledge bases (KBs) appears early on (Petroni et al., 2019). Subsequent research sought to quantify and address inconsistencies in knowledge and reasoning. Efforts include new evaluation protocols (Jang et al., 2021; Laban et al., 2023; Sahu et al., 2022; Feng et al., 2023; Wang et al., 2023) and the development of consistency-aware loss functions (Elazar et al., 2021). These studies highlighted inconsistency not merely as a surface-level artifact, but as a persistent limitation rooted in how LLMs generalize across paraphrased queries. Relevant research has identified improving internal consistency as a key frontier in the development of trustworthy, knowledge-centric LLMs (AlKhamissi et al., 2022).

Parallel work has explored the emergence of reasoning-like behaviors in LLMs, particularly under chain-of-thought (CoT) prompting (Wei et al., 2022). These strategies elicit multi-step answers, raising questions about whether such outputs reflect genuine reasoning or simply surface-level pattern matching (Kojima et al., 2023; Wei et al., 2022). (Wang et al., 2023) specifically studied consistency in CoT-generated answers and proposed strategies for improving it. Comprehensive surveys of reasoning in LLMs (Huang and Chang, 2023; Plaat et al., 2024; Zhang et al., 2024), catalog the current landscape of techniques and open challenges. While much of the existing literature focuses on strategic or contextual reasoning capabilities of LLMs, we argue that foundational inconsistencies arise even at the level of basic conceptual hierarchies. These should be prioritized and systematically examined as a prerequisite to more complex reasoning tasks.

Therefore, we build on the foundational query cluster approach introduced by (Uceda-Sosa et al., 2024), although our work significantly extends this line of inquiry in several ways. First, we adapt and scale the query clustering methodology to a broader set of domains by formalizing domain-specific conceptualization axioms, enabling automated construction benchmarks tailored for industrial applications. Second, we introduce a novel taxonomy of cluster types and corresponding metrics that not only assess the consistency of LLMs, but also expose structural issues within the KGs themselves. Lastly, we release our novel, multi-domain, conceptual consistency dataset.

Crucially, our approach goes beyond simple factual probing by leveraging inter-model consensus to generate domain-specific paraphrases, offering a principled mechanism for evaluating and augmenting both LLM outputs and KG structures. This enables a richer, bidirectional analysis between symbolic and neural representations, improving both the interpretability and trustworthiness of downstream applications.

7 Conclusions and future work

In this work, we have shown that, even when evaluating against a fixed body of knowledge—whether accurate or flawed—state-of-the-art LLMs exhibit between 7–10% inconsistency on basic factual relationships. Notably, our benchmarks contain query clusters of modest size (Table 1), meaning that inconsistencies arise with as few as 4 paraphrased questions. While in-context learning has shown promise in mitigating these inconsistencies (Uceda-Sosa et al., 2024), it does not eliminate them fully.

Addressing this challenge requires further advances in both fine-tuning and prompting strategies. One promising direction involves CoT prompting, with or without explicit instruction (Wei et al., 2023; Wang and Zhou, 2024), which has been shown to improve both consistency and reasoning depth in LLMs. A second avenue for improvement lies in the modeling of conceptual relationships. Future extensions to our framework could incorporate graded membership, contextual reasoning, or type disambiguation, resulting in a more expressive and accurate assessment of model consistency.

Furthermore, large language models often struggle to generalize safely outside of their training distributions. This poses challenges when evaluating consistency against domain-specific knowl-

edge graphs, which typically assume a closed-world semantics, in contrast to the open-world assumptions underlying LLM behavior. This semantic mismatch complicates the interpretation of incompleteness: when a model hedges or abstains from answering, it may reflect uncertainty rather than a true knowledge gap. Bridging this divide will likely require techniques such as uncertainty modeling, retrieval-augmented generation (Lewis et al., 2021), or grounding in structured knowledge sources (Yang et al., 2025).

Altogether, our findings demonstrate that even small, targeted benchmarks can surface meaningful patterns in LLM reasoning behavior. Even further, they can serve as a powerful feedback mechanism to discover problematic subgraphs in reference KGs, offering a novel method for aiding in the curation, maintenance and refinement of domain-specific KGs. Extending this framework to larger knowledge graphs, broader domain coverage, and multi-hop inferential tasks represents a fruitful direction for future work, with the ultimate goal of deploying our method to enable more reliable and trustworthy AI systems.

8 Limitations

While our work presents a principled framework for building benchmarks for evaluating the conceptual consistency of large language models (LLMs) with respect to structured knowledge bases, it is currently limited both in scope and results.

First, despite automating the subgraph extraction process, the initial selection of domains, top-level concepts, and associated properties remains manual. This introduces constraints on scalability and reproducibility, particularly in industrial or proprietary settings where domain-specific knowledge graphs may exhibit idiosyncrasies or unexpected structural complexities. Automating the concept selection process—potentially through ontology alignment or schema matching techniques—could enhance generalization and reduce reliance on manual configuration. Building a community-curated benchmark library spanning multiple domains would also increase robustness, though such an initiative lies beyond the scope of this paper.

Second, our methodology depends on LLMs themselves to generate semantically equivalent paraphrase clusters. As these are shaped by the models' pretraining data, linguistic biases may be introduced—especially in specialized domains where certain formulations are rare or underrepresented. This may limit the semantic coverage of paraphrase clusters. Future work should explore hybrid approaches that incorporate external paraphrasing tools or human-in-the-loop validation to improve semantic fidelity and robustness.

Third, the non-deterministic nature of LLMs poses challenges for consistency evaluation. Even semantically equivalent prompts may yield divergent outputs across multiple runs due to stochastic decoding. While we try to minimize this through cross-model consensus and greedy decoding, other sampling strategies should be explored to further stabilize evaluations and reduce variance.

Still, these limitations suggest promising avenues for future research aimed at improving both the scalability and reliability of LLM conceptual consistency assessment, especially in complex or high-stakes domains.

References

Badr AlKhamissi, Millicent Li, Asli Celikyilmaz, Mona Diab, and Marjan Ghazvininejad. 2022. A review on language models as knowledge bases. *Preprint*, arXiv:2204.06031.

Stefano Borgo, Roberta Ferrario, Aldo Gangemi, Nicola Guarino, Claudio Masolo, Daniele Porello, Emilio Sanfilippo, and Laure Vieu. 2023. Dolce: A descriptive ontology for linguistic and cognitive engineering.

Ronald J Brachman and Hector J Levesque. 2004. *Knowledge Representation and Reasoning*. Elsevier.

Matthew Dahl, Varun Magesh, Mirac Suzgun, and Daniel E Ho. 2024. Large legal fictions: Profiling legal hallucinations in large language models. *Journal of Legal Analysis*, 16(1):64–93.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. Measuring and Improving Consistency in Pretrained Language Models. *Transactions of the Association for Computational Linguistics*, 9:1012–1031.

Fredo Erxleben, Michael Günther, Markus Krötzsch, Julian Mendez, and Denny Vrandečić. 2014. Introducing Wikidata to the Linked Data Web. In *International Semantic Web Conference*, pages 50–65. Springer.

Zhangyin Feng, Weitao Ma, Weijiang Yu, Lei Huang, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting liu. 2023. Trends in integration of knowledge and large language models: A survey and taxonomy of methods, benchmarks, and applications. *Preprint*, arXiv:2311.05876.

Joschka Haltaufderheide and Robert Ranisch. 2024. The ethics of chatgpt in medicine and healthcare: a systematic review on large language models (llms). *npj Digital Medicine*, 7(1):183.

Jie Huang and Kevin Chen-Chuan Chang. 2023. Towards reasoning in large language models: A survey. *Preprint*, arXiv:2212.10403.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

Myeongjun Jang, Deuk Sin Kwon, and Thomas Lukasiewicz. 2021. Accurate, yet inconsistent? consistency analysis on language understanding models. *Preprint*, arXiv:2108.06665.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12):1–38.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2023. Large language models are zero-shot reasoners. *Preprint*, arXiv:2205.11916.

Philippe Laban, Wojciech Kryściński, Divyansh Agarwal, Alexander R. Fabbri, Caiming Xiong, Shafiq Joty, and Chien-Sheng Wu. 2023. Llms as factual reasoners: Insights from existing benchmarks and beyond. *Preprint*, arXiv:2305.14540.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Preprint*, arXiv:2005.11401.

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

Aske Plaat, Annie Wong, Suzan Verberne, Joost Broekens, Niki van Stein, and Thomas Back. 2024. Reasoning with large language models, a survey. *Preprint*, arXiv:2407.11511.

Pritish Sahu, Michael Cogswell, Yunye Gong, and Ajay Divakaran. 2022. Unpacking large language models with conceptual consistency. *Preprint*, arXiv:2209.15093.

Rosario Uceda-Sosa, Karthikeyan Natesan Ramamurthy, Maria Chang, and Moninder Singh. 2024. Reasoning about concepts with llms: Inconsistencies abound. In *Conference on Language Modeling, COLM 2024*.

Jakob Voß. 2016. Classification of knowledge organization systems with wikidata. In *NKOS@TPDL*.

Denny Vrandečić. 2012. Wikidata: A new platform for collaborative data collection. In *Proceedings of the 21st Int. Conf. on world wide web*, pages 1063–1064.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. *Preprint*, arXiv:2203.11171.

Xuezhi Wang and Denny Zhou. 2024. Chain-of-thought reasoning without prompting. *Preprint*, arXiv:2402.10200.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Preprint*, arXiv:2206.07682.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Ziwei Xu, Sanjay Jain, and Mohan Kankanhalli. 2024. Hallucination is inevitable: An innate limitation of large language models. *arXiv preprint arXiv:2401.11817*.

Shiping Yang, Jie Wu, Wenbiao Ding, Ning Wu, Shining Liang, Ming Gong, Hengyuan Zhang, and Dongmei Zhang. 2025. Quantifying the robustness of retrieval-augmented language models against spurious features in grounding data. *Preprint*, arXiv:2503.05587.

Jie Zhang, Haoyu Bu, Hui Wen, Yongji Liu, Haiqiang Fei, Rongrong Xi, Lun Li, Yun Yang, Hongsong Zhu, and Dan Meng. 2025. When llms meet cybersecurity: a systematic literature review. *Cybersecurity*, 8(1):55.

Yadong Zhang, Shaoguang Mao, Tao Ge, Xun Wang, Adrian de Wynter, Yan Xia, Wenshan Wu, Ting Song, Man Lan, and Furu Wei. 2024. Llm as a mastermind: A survey of strategic reasoning with large language models. *ArXiv*, abs/2404.01230.

9 Appendix

9.1 Models used in evaluation

We provide the hugging face URLs for the models used:

- https://huggingface.co/deepseek-ai/DeepSeek-R1

- https://huggingface.co/deepseek-ai/DeepSeek-V2.5
- https://huggingface.co/deepseek-ai/DeepSeek-V3
- https://huggingface.co/ibm-granite/granite-3.0-8b-instruct
- https://huggingface.co/ibm-granite/granite-3.1-8b-instruct
- https://huggingface.co/ibm-granite/granite-3.2-8b-instruct
- https://huggingface.co/meta-llama/Llama-3.1-70B-Instruct
- https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct
- https://huggingface.co/mistralai/Mistral-Large-Instruct-2407
- https://huggingface.co/mistralai/Mixtral-8x22B-Instruct-v0.1
- https://huggingface.co/mistralai/Mixtral-8x7B-Instruct-v0.1

9.2 Wikidata Q and P nodes

Table 3 lists the domains in our released benchmark (as in Table 1) but we also list the Wikidata Q nodes for domains and P nodes for properties.

9.3 Example Semantically Equivalent Queries

To test an edge asserting that A is a subconcept of B, of one such group of semantically equivalent queries to test a single edge, is shown below:

- Is A a subconcept of B?
- Is A a type of B?
- Is every kind of A also a B?
- Is A a subcategory of B?

9.4 Consistent Disagreement

Figure 5 shows how often models consistently disagreed with the reference KG.

Figure 6 shows how often models consistently asserted the existence of an edge that was *not* in the KG.

Domain	Domain Q-Node	Predicate	Property P-node
Academic Disciplines	Q11862829	used for	P366
Dishes	Q746549	has ingredient	P527
Finance Products	Q15809678	used for	P1535
Home Appliances	Q212920	used for	P366
Medical Occupations	Q3332438	has occupation	P425
Music Genres	Q188451	practiced by	P3095
Natural Disasters	Q8065	has cause	P828
Software	Q7397	studied in	P7397

Table 3: Wikidata Q nodes and P nodes for Domains (concepts) and Predicates (properties) respectively.

model	academic disciplines	dishes	Finance	Home Appliances	Medical Specialties	Music Genres	Natural Disasters	Software	AVG
DeepSeek-r1	0.91	0.46	2.9	0.95	2.03	2.2	1.4	0.4	1.41
DeepSeek-V2_5	1.37	0.82	3.17	1.14	1.22	2.29	1.68	1.2	1.61
DeepSeek-V3	1.82	1.09	3.72	1.71	1.89	4.03	1.4	0	1.96
granite-3_0-8b-instruct	2.28	1.55	4.55	1.9	2.03	5.34	1.4	0.8	2.48
granite-3_1-8b-instruct	1.82	1.28	3.59	1.14	1.49	4.26	1.4	1.2	2.02
granite-3_2-8b-instruct	1.82	1.28	3.72	1.14	1.62	4.21	1.4	1.2	2.05
llama-3-1-70b-instruct	0.91	0.59	1.52	0.95	0.95	1.45	1.12	1.41	1.11
llama-3-3-70b-instruct	0.91	0.55	1.59	1.14	1.08	1.78	1.4	1.2	1.21
mistral-large-instruct-2407	1.37	1	3.45	3.24	2.16	3.37	1.4	0.4	2.05
mixtral-8x22B-instruct-v0_1	1.82	0.91	2.9	1.52	1.76	3.37	1.68	0.8	1.85
mixtral-8x7B-instruct-v0_1	1.37	1.09	3.72	2.1	1.62	2.9	2.8	0.8	2.05
AVG	1.46	0.93	3.18	1.52	1.61	3.11	1.54	0.82	1.77

Low consistent disagreement — High consistent disagreement

Low AVG — High AVG

Figure 5: Percentages of consistent disagreement clusters by model and domain.

model	academic disciplines	dishes	finance	home appliances	medical specialties	music genres	natural disasters	software	AVG
DeepSeek-r1	0	0	0.83	0.19	0.41	0.37	0.56	0	0.30
DeepSeek-V2_5	0.23	0	0.55	0.19	0.27	0.7	0.84	1.2	0.50
DeepSeek-V3	0	0	0.41	0.19	0	0.33	0.56	0	0.19
granite-3_0-8b-instruct	0.23	0.09	0.1	0.19	0	0.14	0.28	0.4	0.18
granite-3_1-8b-instruct	0	0	0.31	0.38	0	0	0.84	1.2	0.34
granite-3_2-8b-instruct	0	0	0.16	0.38	0	0	1.12	1.2	0.36
llama-3-1-70b-instruct	0	0.09	0.58	0.19	0.41	0.05	0.56	1.41	0.41
llama-3-3-70b-instruct	0.23	0	0.66	0.19	0.68	0.09	0.56	1.2	0.45
mistral-large-instruct-2407	0	0	0.41	0.19	0.27	0.47	0.84	0.4	0.32
mixtral-8x22B-instruct-v0_1	0.23	0	0.31	0.19	0.41	0.45	1.4	0.8	0.47
mixtral-8x7B-instruct-v0_1	0	0	0.34	0.19	0.14	0.42	0.14	0.8	0.25
AVG	0.08	0.02	0.42	0.22	0.24	0.27	0.70	0.78	0.34

Low hallucination — High hallucination

Low AVG — High AVG

Figure 6: Percentage of edges hallucinated in consistent disagreement clusters