ANSWER SET CONSISTENCY OF LLMS FOR QUESTION ANSWERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Large Language Models (LLMs) can contradict themselves when answering factual questions, especially when asked to enumerate all entities that satisfy a question. We formalize such self-contradiction as *answer-set inconsistency*: given two enumeration questions whose answers satisfy a set-theoretic relation (equivalence, disjointness, containment, etc.), the LLM generates responses violating the relation. To diagnose this phenomenon, we create **Answer-Set Consistency Benchmark (ASCB)**, a benchmark dataset comprising 600 tuples of enumeration questions (2400 questions in total) over which a variety of set-theoretic relations hold, and propose related metrics to quantify answer-set inconsistency. Our evaluation of 18 state-of-the-art LLMs reveals pervasive inconsistency across models, even when the LLM can identify the correct relation. We also analyze likely sources of these failures and use relation-aware prompting as a lightweight exploratory intervention. Making the relation explicit often improves consistency, but gains are uneven and can be offset by abstention or prompt misinterpretation. Our results position answer-set consistency as a complementary dimension for evaluating LLM-based question answering.

1 INTRODUCTION

Large Language Models (LLMs) have demonstrated impressive capabilities not only in natural language understanding and generation, but also in question answering and other tasks involving complex reasoning (Tan et al., 2023; Ding et al., 2024; Li et al., 2024; Saxena et al., 2024; Sui et al., 2024). However, in the context of the latter tasks, they are prone to various types of contradiction (Ghosh et al., 2025; Calanzone et al., 2025) since they are not based on computational techniques that guarantee formal notions of consistency, soundness, etc.

LLMs can also exhibit inconsistency during (factual) question answering. Specifically, in the context of *enumeration questions* requesting a specific list of entities, the responses across different questions may exhibit inconsistency with respect to set-theoretic relations between them. Take, for example, the four questions in Table 1. All such questions are enumeration questions: they expect a set of entities as answers. Let $[\![Q]\!]$ denote the expected set of answers for a question Q. Among these four questions, we can see that certain set-theoretic relations should be expected to hold, including *equality* ($[\![Q_1]\!] = [\![Q_2]\!] = [\![Q_3]\!] \cup [\![Q_4]\!]$), *containment* ($[\![Q_3]\!] \subseteq [\![Q_1]\!], [\![Q_4]\!] \subseteq [\![Q_1]\!], [\![Q_3]\!] \subseteq [\![Q_2]\!], [\![Q_4]\!] \subseteq [\![Q_2]\!]$, and such relations entailed by equality), *disjointness* ($[\![Q_3]\!] \cap [\![Q_4]\!] = \emptyset$), etc. However, the answers returned by a particular model may not satisfy these relations, even in cases where the model can recognize the correct expected relation. Specifically, let $[\![Q]\!]_M$ denote the set of answers enumerated by model M. Then, for example, given Q_1 and Q_3, when asked what relation holds between their answers, M may correctly recognize the containment of the latter in the former ($[\![Q_3]\!] \subseteq [\![Q_1]\!]$), but still enumerate $[\![Q_1]\!]_M$ and $[\![Q_3]\!]_M$ such that $[\![Q_3]\!]_M \not\subseteq [\![Q_1]\!]_M$, thus contradicting itself.

We formalize this issue as *answer-set inconsistency*, wherein *the answers for a tuple of factual enumeration questions generated by a particular model do not satisfy the set-theoretic relations that are expected to hold for that tuple*. We further consider an *answer-set contradiction* whereby *the model enumerates answers that do not satisfy the set-theoretic relation it itself predicts*. Related topics have been well-studied in database theory literature wherein the notions of *query containment*, *equivalence*, etc., are textbook topics (Abiteboul et al., 1995), with decades of theoretical and practical results covering a variety of query languages, data models, and reasoning formalisms. Unlike database

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Table 1: Illustrative example with four enumeration questions forming different relationships (equivalence, containment, disjointness, etc.) with respect to the expected answer sets.

Q_1	What are the tributaries of the Madeira River?
Q_2	Which rivers & streams flow directly into the Madeira River?
Q_3	What are the right-bank tributaries of the Madeira River?
Q_4	What are the left-bank tributaries of the Madeira River?

systems, generative Artificial Intelligence (AI) models are not designed to guarantee satisfaction of such formal relations in the answers they provide. Perhaps for this reason, answer-set (in)consistency has not been well-studied in the context of generative AI. To the best of our knowledge, the closest work is that of Elazar et al. (2021), which evaluates consistency across paraphrased versions of cloze-style phrases, but these permit only a single answer. Hogan et al. (2025) briefly discuss this issue as "*coherence*", but do not investigate it further. Given that such models are increasingly used to answer users' enumeration questions, we believe the topic merits more analysis.

Research questions. In this paper, we address the following research questions (RQs):

RQ1: To what extent do LLMs produce (in)consistent answer sets for enumeration questions?

RQ2: Can LLMs recognize the set-theoretic relations that exist between enumeration questions?

RQ3: Which set-theoretic relations cause the most answer-set inconsistency in LLMs?

RQ4: What are the key factors that cause answer-set inconsistency in LLMs?

RQ5: To what extent does relation-aware prompting affect answer-set consistency?

Contributions. This paper presents three main contributions: (1) We highlight and formalize the notion of answer-set consistency for enumeration questions. (2) To quantify answer-set consistency, we develop and release a novel handcrafted benchmark of 600 question quadruples, with 2,400 questions in total, where fixed relations are expected to hold between the questions of each tuple. We further propose measures to quantify answer-set consistency with respect to such a dataset. (3) We present an empirical analysis of 18 state-of-the-art LLMs on the benchmark.

Our methodology allows for a systematic analysis of how model size, architecture, and prompting strategy affect the logical coherence of LLM outputs for question answering. Our findings reveal that LLMs exhibit significant answer-set inconsistency, the extent of which depends on the particular model and the particular relation (surprisingly, newer or bigger models do not universally outperform older or smaller variants). Such inconsistency often occurs even when the model is able to correctly recognize the relation expected to hold. We further include strategies showing that making the relation explicit generally improves answer-set consistency, though asking the model to infer the relation itself can sometimes increase abstention or introduce prompt-misinterpretation errors.

The code, datasets, and experimental results are available in the anonymous GitHub repository (authors, 2026a); the dataset is also available on Hugging Face (authors, 2026b).

2 Related Work

Question answering datasets. A wide range of datasets target different QA settings Tan et al. (2023); Wang et al. (2025); Lee & Kim (2024); Singhal et al. (2024); Iqbal et al. (2024); Zheng et al. (2024); Zhou & Duan (2024); Zhu et al. (2024); Allemang & Sequeda (2025); Ma et al. (2025), typically evaluating accuracy against predefined ground truth. While essential for assessing correctness and completeness, such benchmarks do not capture internal model consistency, which we study as a complementary dimension.

Consistency of LLMs. Prior work explores multiple notions of consistency. Ghosh et al. (2025) study logical consistency in boolean fact-checking, evaluating negation, disjunction, and conjunction, and show improvements with model scale. Calanzone et al. (2025) enforce logical constraints via neuro-symbolic fine-tuning, outperforming standard approaches. Bansal & Gangwani (2026) study logical consistency across contrapositive reasoning, transitivity, syllogisms, negation, modus tollens,

and commonsense entailment, using a set-level metric that requires all questions in a set to be correct and a Natural Language Inference-based method to detect and repair contradictions.

Beyond propositional logic, Liu et al. (2025) examine preference consistency (e.g., transitivity, order-invariance), finding frequent violations correlated with downstream performance. Jang et al. (2022) categorize consistency into semantic, logical, and factual dimensions. Factual consistency has been studied via paraphrase-based evaluation Elazar et al. (2021) and cross-examination between models Cohen et al. (2023). Related work considers equivalence, e.g., SQL query equivalence Zhao et al. (2025) and program equivalence benchmarks Wei et al. (2025), but focuses on symmetric relations.

Query containment. Given a database D and query Q, let $[\![Q]\!]_D$ denote its answers. Query containment asks whether $[\![Q_1]\!]_D \subseteq [\![Q_2]\!]_D$ for all D, while equivalence requires equality (Abiteboul et al., 1995). These problems are decidable for restricted languages (e.g., conjunctive queries) but undecidable for more expressive ones (e.g., relational algebra, full SQL). Thus, while LLMs cannot solve equivalence in general, it is important to characterize their limits and improve performance.

While prior work studies consistency for boolean statements, facts, paraphrases, and joint correctness across related queries, we study the unexplored consistency of set-valued answers through set-theoretic relations, without requiring ground-truth answer sets.

3 Answer-Set Consistency

This section defines answer-set consistency and presents the dataset, metrics, and prompting conditions used in our evaluation.

3.1 Definition

We say that Q is an *enumeration question* if it has a ground truth answer that is a set, denoted $[\![Q]\!]$ (see Table 1). Let M be a model (e.g., an LLM, potentially primed with a suitable prompt). We denote by $[\![Q]\!]_M$ the answer set generated by model M for question Q.

We now define the related notions of *answer-set consistency* and *answer-set contradiction*.

Given two enumeration questions Q_1 and Q_2 that are equivalent (i.e., $[\![Q_1]\!] = [\![Q_2]\!]$), we say that a model M is *answer-set consistent* with respect to the equivalence of (Q_1,Q_2) if and only if $[\![Q_1]\!]_M = [\![Q_2]\!]_M$. If this property does not hold, we call the model *answer-set inconsistent*. We extend answer-set (in)consistency to the other standard binary set relations, including *containment* (given $[\![Q_1]\!] \subseteq [\![Q_2]\!]$, check $[\![Q_1]\!]_M \subseteq [\![Q_2]\!]_M$), *disjointness* (given $[\![Q_1]\!] \cap [\![Q_2]\!] = \emptyset$, check $[\![Q_1]\!]_M \cap [\![Q_2]\!]_M = \emptyset$) and *overlap* (given $[\![Q_1]\!] \cap [\![Q_2]\!] \neq \emptyset$, check $[\![Q_1]\!]_M \cap [\![Q_2]\!]_M \neq \emptyset$). Notice that we do not need ground-truth answer sets for questions in order to analyze answer-set consistency.

Example 3.1. Considering the four questions in Table 1, even without the ground-truth answer sets, we see that $[\![Q_1]\!] = [\![Q_2]\!]$ (equivalence), since *tributaries* of a river are defined as *rivers and streams that flow directly into that river*. Similarly, we see that $[\![Q_3]\!] \subseteq [\![Q_1]\!]$ and $[\![Q_4]\!] \subseteq [\![Q_1]\!]$ (containment) since both left- and right-bank tributaries should be included in the answer for all tributaries. Furthermore, we see that $Q_3 \cap Q_4 = \emptyset$ (disjointness), as right-bank and left-bank tributaries are mutually exclusive. Other implied relations exist, for example, $[\![Q_3]\!] \subseteq [\![Q_2]\!]$ (containment), $[\![Q_3]\!] \cap [\![Q_2]\!] \neq \emptyset$ (overlap). Hence, for example, if a model would give overlapping answers for Q_3 and Q_4, we would deem it answer-set inconsistent with respect to the disjointness of (Q_3, Q_4).

The presented definition is agnostic to the open world assumption, or closed world assumption, since we are solely interested in the relationship between two answers, regardless of whether elements not included in the answer are treated as false, or unknown to the model. However, under the open world assumption, a model may return empty answers, which are trivially consistent; for this reason, in our experiments, we will handle cases in which the models returns empty answers separately.

Next, we define *answer-set contradictions*, a fuller treatment is deferred to Appendix K. For two enumeration questions Q_1 and Q_2, let $[\![Q_1, Q_2]\!]_M \subseteq \{\text{E}, \text{C}, \text{D}, \text{O}\}$ denote the relations that model M predicts between (Q_1, Q_2), corresponding to equivalence, containment, disjointness, and overlap. We use a set because multiple relations may hold simultaneously. Model M produces an *answer-set contradiction* if there exists $R \in [\![Q_1, Q_2]\!]_M$ that does not hold between $[\![Q_1]\!]_M$ and $[\![Q_2]\!]_M$.

That is, the model's predicted relation is not satisfied by its own answers, yielding an internal self-contradiction rather than an inconsistency with respect to a gold-standard relation.

Example 3.2. Considering the first two questions in Table 1, if a model predicts that these two questions are equivalent (E), but gives answer sets $[\![Q_1]\!]_M$ and $[\![Q_2]\!]_M$ such that $[\![Q_1]\!]_M \neq [\![Q_2]\!]_M$, this represents an answer-set contradiction with respect to equivalence.

3.2 DATASET

To evaluate the *answer-set consistency* of various LLMs and address our research questions, we require related enumeration questions whose answer sets satisfy known equivalence, containment, and disjointness relations. Given that no existing dataset provides these relations at sufficient scale, we construct one according to the following standards and procedures.

Design standards. We retain a question only if it (i) requests an unordered set of entities rather than a Boolean, scalar, or free-form response; (ii) has an objective, non-empty, and well-delimited answer set; (iii) yields between 2 and 100 answers; and (iv) is fluent and unambiguous in English. The lower cardinality bound permits strict, non-empty containment, while the upper bound limits context-window pressure for smaller models. We additionally seek diversity across topics and exclude questions whose membership criterion is inherently subjective or vague. For example, *Which countries are members of the European Union?* defines a crisp set, whereas *What are the jobs in finance?* does not.[1]

Candidate sourcing and screening. We source candidates from three KGQA datasets—LC-QUAD 2.0 (Dubey et al., 2019), QALD (Usbeck et al., 2024), and QAWIKI (Moya Loustaunau & Hogan, 2025)—because their associated structured queries allow us to screen for enumeration form and cardinality. Available paraphrases provide candidates for equivalence, although only QAWIKI contains explicit containment examples, and these are limited in number. For LC-QUAD 2.0 and QALD, an LLM-assisted pipeline first filters unsuitable questions and proposes relation-constrained companions; Appendix G.2 specifies the complete pipeline and prompts. We additionally construct a SYNTHETIC source using questions generated from scratch by LLMs; its generation procedure is detailed in Appendix G. In many cases, source or generated questions served only as inspiration for the final quadruple and were substantially modified by hand for inclusion in the benchmark. These outputs form a candidate pool rather than automatically accepted benchmark items.

Relation-constrained construction. Each retained item follows the same four-question template (Q_1, Q_2, Q_3, Q_4). We require $[\![Q_1]\!] = [\![Q_2]\!]$, $[\![Q_3]\!] \subsetneq [\![Q_1]\!]$, $[\![Q_4]\!] = [\![Q_1]\!] \setminus [\![Q_3]\!]$. Thus, Q_1 and Q_2 are semantically equivalent and therefore have the same answer set; Q_3 and Q_4 each denote a non-empty strict subset of the broader answer set; and Q_3 and Q_4 form an exhaustive, mutually exclusive partition of that set. LLMs were used to propose Q_2–Q_4 when suitable source questions were unavailable, and the required relations determined whether a proposal was retained.

Each quadruple induces 12 directed primary relations. Equivalence holds for (Q_1, Q_2) and (Q_2, Q_1). We call superset containment *broader*; it holds for (Q_1, Q_3), (Q_1, Q_4), (Q_2, Q_3), and (Q_2, Q_4). Its inverse, *narrower*, holds for (Q_3, Q_1), (Q_3, Q_2), (Q_4, Q_1), and (Q_4, Q_2). Finally, disjointness holds for (Q_3, Q_4) and (Q_4, Q_3). The template also entails secondary relations: equivalence implies containment in both directions, while each non-empty subset overlaps with its broader set. Table 1 presents an example quadruple satisfying this specification.

Manual verification and quality control. Three authors participated in the manual curation process. Each quadruple retained in **ASCB** was independently verified by at least two authors against the predefined design standards and formal relation constraints. This verification was substantive: source questions often served only as starting points, and LLM-proposed complements frequently

[1] Here, "crisp" means that each candidate entity can be assigned unambiguously to either the answer set or its complement.

required correction because they did not form a true partition.[2] LLM-based suggestions for phrasing and error correction were likewise adopted only after manual review.

The final **ASCB** contains 150 English question quadruples from each of the four sources, for 600 quadruples (2,400 questions) in total. The benchmark is available on Hugging Face (authors, 2026b); licensing and source attribution are detailed in Appendix G.3.

3.3 Evaluation Tasks and Prompting Conditions

To evaluate the answer-set consistency of LLMs using our **ASCB**, we define three distinct tasks, two of which test whether making set relations explicit affects consistency. Rather than exploring the full spectrum of possible mitigation approaches, this work focuses on lightweight conversational strategies for making set relations explicit, allowing us to test when such relation information improves consistency and when it introduces failure modes such as abstention. While alternative mitigation strategies (e.g., fine-tuning Ghosh et al. (2025); Calanzone et al. (2025)) exist, we leave their analysis for future work as they often incur substantial constraints, such as high computational cost, reliance on high-quality training data, limited applicability to closed-source models, and trade-offs with general task performance. For all tasks, the LLMs under evaluation are configured with the lowest possible temperature. The prompts used are reported in Appendix F.

Task 1: Base evaluation. Our first task evaluates the base answer-set consistency of the LLM under test, and involves two subtasks.

In Task 1.1, *Classification*, the models are presented pairs of questions (Q_i, Q_j) taken from each quadruple and asked to identify the *first* set-theoretic relation that holds, in the following order, from the answer set of Q_i to that of Q_j: equivalence ($[\![Q_i]\!] = [\![Q_j]\!]$), contained by ($[\![Q_i]\!] \subseteq [\![Q_j]\!]$), contains ($[\![Q_i]\!] \supseteq [\![Q_j]\!]$), disjointness ($[\![Q_i]\!] \cap [\![Q_j]\!] = \emptyset$) and overlap ($[\![Q_i]\!] \cap [\![Q_j]\!] \neq \emptyset$). Although any pair of questions must satisfy at least one such relation, we further add an *unknown* option (‘`idk`’) in case the model cannot confidently determine the relation.

In Task 1.2, *Enumeration*, all 2,400 questions are posed to the models independently, each within its own isolated context, requesting the LLM to enumerate all answers for the question. The prompt further instructs the model to return an exhaustive list separated by ‘|’, to avoid additional text, to return ‘`idk`’ in case it cannot answer, to return ‘`no answer`’ if there is no answer, and to expand acronyms and use full names whenever possible. The corresponding answers are then collected.

Task 2: Classification-then-Enumeration (CtE). We further evaluate a relation-aware prompting condition that operates within a conversational context. Specifically, the model is first asked to identify the relationship between the questions and is then prompted to enumerate their answers. This condition tests whether explicitly asking the model to identify the relation before enumeration affects answer-set consistency. In this sense, the method functions as a form of Chain-of-Thought prompting (Wei et al., 2022): the model is encouraged to first reason about the relevant relationship and then answer the questions accordingly.

Task 3: Oracle. We first run Task 1.1, and if we detect an answer-set inconsistency with respect to a given relation for a question pair (Q_i, Q_j), we tell the LLM the correct relation that holds, and request it to enumerate answers again. This task is an ideal version of Task 2, as it assumes an oracle that knows what relation holds between the questions. This will give us insights into what the model could achieve in Task 2 if it always classifies the relation correctly.

Question and relations considered. Our quadruples provide a large number of potential binary relations to test, but many such relations are redundant. Thus, we only consider each pair of questions once (skipping symmetric and inverse relations), and only test for the primary relation. The selected relations are presented in Table 2. The relations $E_{1,2}$, $N_{3,1}$, $N_{4,1}$, and $D_{3,4}$ cover equivalence, containment (narrower) and disjointness. We also test an n-ary relation $E_{4,1\setminus 3}$ involving set difference as an example of a more complex task. Finally, to control for non-determinism of the model over time, we test $E_{1,*}$, which runs the same question Q_1 at different times in fresh contexts.

[2] For example, *What counties of North Dakota use the Mountain Time Zone?* and *What counties of North Dakota use a time zone other than Mountain Time?* are not necessarily complementary because a county may span multiple time zones.

Table 2: The relations considered for experiments: their notation, definition and meaning

Notation	Definition	Meaning
$E_{1,2}$	$[\![Q_1]\!] = [\![Q_2]\!]$	*Equivalence*: Results of Q_1 are the same as those of Q_2
$N_{3,1}$	$[\![Q_3]\!] \subseteq [\![Q_1]\!]$	*Narrower*: Results of Q_3 are contained in those of Q_1
$N_{4,1}$	$[\![Q_4]\!] \subseteq [\![Q_1]\!]$	*Narrower*: Results of Q_4 are contained in those of Q_1
$D_{3,4}$	$[\![Q_3]\!] \cap [\![Q_4]\!] = \emptyset$	*Disjointness*: No result of Q_3 is a result of Q_4
$E_{4,1\setminus 3}$	$[\![Q_4]\!] = [\![Q_1]\!] \setminus [\![Q_3]\!]$	*Equivalence*: Results obtained by removing the answer set of Q_3 from that of Q_1 are the same as those of Q_4
$E_{1,*}$	$[\![Q_1]\!] = [\![Q_1^*]\!]$	*Equivalence*: Results of Q_1 are the same as those of Q_1 posed in a different context at a different time

3.4 Evaluation Measures

We present various measures to evaluate the answer-set consistency of LLMs with respect to the previous tasks. The first addresses performance for classification, while the following two address performance for enumeration.

Classification accuracy. We first quantify the ability of LLMs to correctly classify the set-theoretic relation that holds between the answer-sets of the questions. For each relation $R \in \{E_{1,2}, N_{3,1}, N_{4,1}, D_{3,4}, E_{4,1\setminus 3}\}$, the *classification accuracy* of a model M over n test instances ($n = 600$ for **ASCB**) is defined as $R^{\text{ACC}}(M) = \frac{\text{\# correct classifications of } R \text{ by } M}{n}$. We further denote by $\text{ACC}(M)$ the average of the accuracy of the five aforementioned relations for the model M.

Consistency rates. For each pair of enumerated answer sets, we evaluate whether or not the six expected relations $R \in \{E_{1,2}, N_{3,1}, N_{4,1}, D_{3,4}, E_{4,1\setminus 3}, E_{1,*}\}$ are satisfied. Each such binary relation R is checked independently over n test instances, where we say that an instance is consistent for R if the enumerated answers for the questions satisfy R. The *consistency rate* of M for R is then defined as $R^{\text{CON}}(M) = \frac{\text{\# consistent answer sets for } R \text{ by } M}{n}$. Empty answer sets are excluded from these while '`idk`' responses are counted as incorrect for all metrics involving the corresponding question and are reported separately. We further denote by $\text{CON}(M)$ the average consistency rate of M over the five relations excluding $E_{1,*}$ (which is intended as a control).

Jaccard similarity. The consistency rates consider each $(Q_i, Q_i', \star_i)$ as a discrete result (1 for correct, or 0 for incorrect), irrespective of how close the answer sets of the two questions are to satisfying the relation. To complement the first measure, we consider a continuous measure that captures the degree to which an instance is satisfied. Specifically, we use the well-known *Jaccard similarity*, defined for sets S_1 and S_2 as $J(S_1, S_2) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|}$. For a given model M, and test instance (Q_i, Q_j), we define the similarity of the test instance as $J([\![Q_i]\!]_M, [\![Q_j]\!]_M)$. The Jaccard similarity of a model M for a relation $R \in \{E_{1,2}, D_{3,4}, E_{4,1\setminus 3}, E_{1,*}\}$ over n test instances is then defined as $R^{\text{SIM}}(M) = \frac{\text{sum of similarity of all test instances by } M}{n}$, i.e., as the average similarity over R. The '`idk`' responses are counted as incorrect for all metrics involving the corresponding question and are reported separately. We use this measure for equivalence ($E_{1,2}$, $E_{4,1\setminus 3}$, $E_{1,*}$) and disjointness ($D_{3,4}$) relations, where a score close to 1 is good in the former case, and poor in the latter case.

Empty rates. We denote by $\%\text{IDK}$ the percentage of instances giving '`idk`' or empty answers.

Hypotheses and significance testing. We propose two hypotheses: (H_1) The **CtE** and **Oracle** conditions yield less answer-set inconsistency than **Base**. (H_2) LLMs with better general performance produce more consistent responses. To test statistical significance, we apply the one-sided McNemar test (Lachenbruch, 2014), suitable for paired nominal data. We use $\alpha = 0.05$ for significance. The null hypothesis of no improvement is rejected if the one-sided p-value satisfies $p < 0.05$.

Stochasticity control. The causes of answer-set inconsistency in LLMs can be attributed to two main factors. (1) *Stochasticity* in the generation process, such as during token sampling, or latent variability, leading the model to produce different outputs for the same query across runs. (2) *Semantic misunderstanding*, where the model misinterprets or overlooks the logical or semantic relations among queries, resulting in violations of restrictions. We introduce $E_{1,*}$ precisely to estimate and control

for the effects of (1), which varies across models. Further discussion of the causes behind LLM inconsistencies can be found in Appendix L.

The impact specifically of the two factors on the answer-set inconsistency of model M for $E_{1,2}$ can be assessed by comparing $E^{\text{CON}}_{1,2}(M)$ with $E^{\text{CON}}_{1,*}(M)$ and $E^{\text{SIM}}_{1,2}(M)$ with $E^{\text{SIM}}_{1,*}(M)$: a large gap indicates (2) plays a smaller role, and a small gap indicates (2) plays a larger role. Conversely, we can compare the difference of consistency for equivalence relations like $E_{1,*}$, $E_{1,2}$ with other relations like $N_{3,1}$, $N_{4,1}$, $D_{3,4}$ where (assuming stochasticity plays a similar role for such relations) a large gap indicates (2) plays a larger role, and a small gap indicates (2) plays a smaller role.

4 RESULTS

We report experimental results for 18 LLMs , with the temperature set to the minimum possible value (zero, when supported) for all tasks. For answer-set consistency, results are shown in Table 3, while for the classification task we summarize the main findings here, with the full table in Appendix I. Models are ordered by their Global Average scores from White et al. (2025) (A = worst, R = best), and models not in the benchmark are positioned alongside models of similar size. Results are reported on the aggregated dataset described in Section 3.2, with main trends in Appendix H and individual dataset results available on GitHub (authors, 2026a).

Classification task. Appendix I reports LLM accuracy on the relation classification task, with the highest scores in each column highlighted in bold. Results show substantial variability: smaller models (e.g., Llama-3.1-8b) perform poorly, often below 20%, while GPT-oss-20b, GPT-4.1-nano, and Mistral-small:24b show inconsistent performance across relations. Larger models, including Gemini-2.5-pro, GPT-5-nano, GPT-5, GPT-o3, and Grok-3-mini, consistently reach accuracies near or above 90%. Overall, $E_{1,2}$ and $D_{3,4}$ are the easiest relations, whereas $N_{4,1}$ remains the most challenging. Gemini-2.5-pro achieves the best overall performance, closely followed by GPT-5.

Answer-Set Consistency. Table 3 lists the results for answer-set consistency, considering the control relation, all five test relations, and the measures of consistency rate and Jaccard similarity. Recall that, exceptionally for $D^{\text{SIM}}_{3,4}$, lower values are better, since the answer sets are expected to be disjoint. In Table 3, '`idk`' responses are counted as incorrect for all metrics involving the corresponding question. For example, if a model answers '`idk`' for (Q_1), we assign the lowest consistency score to every relation depending on (Q_1). This prevents abstention from artificially inflating consistency by restricting evaluation to the few instances where the model provides concrete answers, or by avoiding one side of a relation. Counting '`idk`' as incorrect therefore provides a conservative estimate of answer-set consistency. To account for the alternative view that this may penalize cautious models, Appendix A reports results with '`idk`' responses filtered out, while Appendix B presents an ablation in which the '`idk`' option is removed from the prompt instructions.

We see high rates of answer-set inconsistency for all relations, including even the control relation. The small gap between $E_{1,*}$ and $E_{1,2}$ suggests that stochasticity plays an important role in answer-set inconsistency, even though temperature was lowered as much as possible in all such cases. This phenomenon aligns with prior research demonstrating that LLMs produce inconsistent outputs even with identical inputs. The literature identifies four primary sources of this variability: decoding randomness (Ackley et al., 1985; Li et al., 2025; Atıl et al., 2025; Renze, 2024), computational nondeterminism (Yuan et al., 2025; Masoudnia & Ebrahimpour, 2014; Dao et al., 2022; Atıl et al., 2025), order sensitivity (Vaswani et al., 2017; Liu et al., 2024a; Lu et al., 2022), and data-level conflicts (Xu et al., 2024; Nakshatri et al., 2025; Xie et al., 2024). Beyond inherent stochasticity, our error analysis of LLM responses under the **CtE** and **Oracle** conditions revealed recurring patterns: inconsistent terminology (despite instructions to avoid these cases, e.g., "Spain" vs. "Kingdom of Spain"), variations in answer completeness, failure to capture implicit logic, and absence of an uncertainty option, which increases hallucinations and reduces consistency. To quantify the impact of terminological variation, we further used Wikidata (Vrandecic & Krötzsch, 2014) to disambiguate entities (Appendix D). Additional error-analysis details appear in Appendices L and M. The remaining gap between $E_{1,2}$, $E_{1,*}$, and containment/ternary relations suggests that semantic misunderstandings are also a major source of inconsistency.

In terms of binary consistency rates (CON), the most challenging relation is the ternary relation $E_{4,1\setminus 3}$, while disjointness $D_{3,4}$ achieves the highest in the **CtE** setting. Under the **Base** strategy, the

Table 3: Per-relation consistency (0-100 scale) for each model and strategy (Str).

ID	Model	Str	$E^{\text{CON}}_{1,*}$	$E^{\text{SIM}}_{1,*}$	$E^{\text{CON}}_{1,2}$	$N^{\text{CON}}_{3,1}$	$N^{\text{CON}}_{4,1}$	$D^{\text{CON}}_{3,4}$	$E^{\text{CON}}_{4,1\setminus 3}$	CON	$E^{\text{SIM}}_{1,2}$	$D^{\text{SIM}}_{3,4}$	$E^{\text{SIM}}_{4,1\setminus 3}$	%IDK
A	Llama-3.1-8b	Base	3.26	21.88	4.00	11.83	7.50	67.67	0.00	18.20	25.71	9.67	10.12	0.75
		CtE	–"–	–"–	16.33	9.83	33.83	68.17	5.00	26.63	56.03	9.74	38.99	**0.08**
		Ora.	–"–	–"–	12.67	15.50	12.33	61.33	1.83	20.73	45.49	11.84	23.97	0.21
B	GPT-oss-20b	Base	13.32	35.05	14.33	30.50	21.67	64.33	7.83	27.73	34.05	26.03	23.02	14.84
		CtE	–"–	–"–	56.00	30.33	57.83	67.33	36.00	49.50	68.23	28.06	52.44	15.75
		Ora.	–"–	–"–	51.67	66.33	72.67	75.33	52.33	63.67	69.84	23.97	69.64	11.92
C	GPT-4.1-nano	Base	6.80	17.27	23.67	32.50	30.33	41.33	2.17	26.00	48.83	39.59	18.88	13.79
		CtE	–"–	–"–	0.67	0.00	0.00	10.00	0.00	2.13	0.82	89.66	0.00	79.83
		Ora.	–"–	–"–	43.33	40.33	48.00	36.17	12.50	36.07	55.21	48.60	26.77	18.58
D	Mistral-small:24b	Base	19.06	35.29	12.88	13.39	13.05	22.37	0.51	12.44	23.68	62.94	7.85	38.69
		CtE	–"–	–"–	73.67	41.17	70.17	68.00	41.17	58.84	81.73	29.58	56.68	11.42
		Ora.	–"–	–"–	62.17	60.17	73.67	68.00	35.17	59.84	73.13	29.43	54.31	15.33
E	Llama-3.1-70b	Base	20.33	48.82	16.33	26.50	21.83	64.83	2.17	26.33	40.57	15.95	18.39	2.75
		CtE	–"–	–"–	79.17	37.67	85.50	**90.00**	55.83	69.63	89.54	**6.45**	79.78	2.37
		Ora.	–"–	–"–	60.33	69.00	78.33	83.33	47.50	67.70	74.62	13.76	71.01	4.54
F	Gemini-2.0-flash	Base	**42.86**	**64.83**	32.67	41.50	27.33	53.50	3.83	31.77	60.08	22.38	25.39	3.00
		CtE	–"–	–"–	73.67	39.17	62.83	57.17	48.17	56.20	80.17	35.97	57.79	22.46
		Ora.	–"–	–"–	78.67	83.50	84.33	87.67	68.33	80.50	82.00	10.45	**83.21**	2.96
G	GPT-4.1-mini	Base	21.25	47.09	16.33	38.50	14.17	56.00	2.00	25.40	34.56	26.98	15.11	8.21
		CtE	–"–	–"–	74.17	38.50	68.50	71.17	60.33	62.53	80.06	27.83	65.79	18.54
		Ora.	–"–	–"–	73.83	80.17	85.83	73.50	58.67	74.40	86.14	22.39	71.85	8.63
H	GPT-4o	Base	19.86	36.96	7.67	10.83	6.50	4.50	1.50	6.20	9.32	93.93	3.04	71.08
		CtE	–"–	–"–	40.67	27.33	31.83	33.33	27.17	32.07	41.84	65.79	29.22	59.83
		Ora.	–"–	–"–	55.50	54.67	58.00	57.00	43.83	53.80	62.09	41.80	51.54	32.33
I	GPT-4.1	Base	25.46	56.30	30.33	47.67	35.33	54.17	9.00	35.30	60.38	20.43	35.29	3.75
		CtE	–"–	–"–	**86.67**	52.50	**86.00**	82.83	74.17	76.43	**92.38**	15.90	82.32	8.16
		Ora.	–"–	–"–	78.33	**85.83**	84.50	84.67	67.00	80.07	89.24	13.76	81.23	6.42
J	Grok-3-mini	Base	30.00	55.43	26.17	47.33	34.17	73.00	18.33	39.80	52.42	19.84	39.45	8.54
		CtE	–"–	–"–	79.67	49.17	75.83	78.17	65.00	69.57	85.22	21.27	73.07	13.66
		Ora.	–"–	–"–	79.17	80.50	77.17	73.67	64.17	74.94	85.47	25.83	72.18	13.25
K	DeepSeek-V3.1	Base	27.69	49.86	28.83	29.50	27.17	31.50	3.00	24.00	54.03	42.48	20.13	12.67
		CtE	–"–	–"–	72.83	53.17	61.67	58.50	51.33	59.50	75.55	39.85	56.94	27.21
		Ora.	–"–	–"–	76.33	77.83	81.50	76.00	59.00	74.13	86.34	22.63	72.99	9.79
L	Gemini-2.5-flash	Base	34.66	63.10	31.17	44.50	26.67	64.83	14.83	36.40	59.68	25.72	32.46	8.58
		CtE	–"–	–"–	84.33	50.33	82.17	81.67	**76.00**	74.90	89.00	17.82	78.56	10.00
		Ora.	–"–	–"–	85.17	83.00	82.50	84.50	**76.00**	**82.23**	89.06	15.25	80.99	7.84
M	GPT-5-nano	Base	14.09	22.79	23.67	30.50	26.83	40.33	15.17	27.30	37.12	57.62	26.30	43.50
		CtE	–"–	–"–	71.67	5.50	67.67	14.00	0.50	31.87	72.00	85.89	5.68	7.17
		Ora.	–"–	–"–	47.50	45.17	45.83	46.33	35.83	44.13	53.31	52.90	42.07	41.12
N	DeepSeek-R	Base	22.27	44.59	17.50	37.33	26.17	75.00	10.00	33.20	45.77	10.96	30.90	4.21
		CtE	–"–	–"–	64.67	43.67	53.50	57.50	47.67	53.40	68.57	41.95	50.58	23.75
		Ora.	–"–	–"–	65.83	69.33	78.83	83.17	53.33	70.10	80.21	15.31	74.54	5.92
O	Gemini-2.5-pro	Base	29.72	57.68	29.50	44.67	30.33	75.17	16.33	39.20	62.86	11.21	39.95	3.29
		CtE	–"–	–"–	80.00	49.00	74.67	80.17	67.00	70.17	84.94	19.05	71.58	11.08
		Ora.	–"–	–"–	72.83	67.33	75.50	87.00	62.67	73.07	80.13	12.90	73.83	8.25
P	GPT-5-mini	Base	20.14	31.99	18.33	26.67	22.17	29.33	14.50	22.20	26.71	68.57	21.09	55.12
		CtE	–"–	–"–	50.67	31.33	33.83	53.17	30.17	39.83	55.24	46.83	31.76	39.71
		Ora.	–"–	–"–	40.50	38.67	42.33	41.67	35.00	39.63	45.04	58.25	38.05	49.38
Q	GPT-o3	Base	25.03	52.78	21.36	40.85	28.81	76.10	15.93	36.61	47.49	17.79	36.67	9.83
		CtE	–"–	–"–	74.33	45.50	79.83	84.67	70.83	71.03	85.57	14.57	77.35	9.62
		Ora.	–"–	–"–	60.67	77.50	76.00	82.33	64.83	72.27	78.18	17.44	75.12	13.38
R	GPT-5	Base	31.16	49.80	21.00	40.33	28.67	50.17	18.17	31.67	35.26	46.20	32.60	37.50
		CtE	–"–	–"–	65.83	44.33	60.33	61.83	56.50	57.76	70.13	38.03	58.28	30.29
		Ora.	–"–	–"–	57.67	60.17	59.00	58.83	53.83	57.90	63.61	41.17	57.11	34.54

best model in terms of average consistency is Grok-3-mini ($\sim 39\%$). By contrast, the worst model, GPT-4o, exhibits much lower average consistency ($\sim 6\%$).

In the **Base** setting, consistency for $N_{3,1}$ is consistently better than for $N_{4,1}$. Because Q_4 expresses the complementary restriction, its wording and answer-set composition may contribute to this difference. An additional lexical audit finds explicit negation or exclusion markers in 399 of 600 Q_4 questions,

compared with 38 of 600 Q_3 questions. However, the difference in **Base** containment consistency between marked and unmarked Q_4 questions is not statistically significant (Appendix E). Thus, the observed asymmetry cannot be attributed to explicit negation on the basis of this analysis. Wording, question content, and answer-set composition remain possible contributors.

CtE improves consistency over **Base** for $17/18$ models in the average consistency score, with GPT-4.1-nano being the only model whose CON score decreases under **CtE**, largely due to a sharp increase in abstention. Improvements can be explained by the high classification accuracy for these relations (see Appendix I) and by the fact that **CtE** encourages the LLM to reason explicitly about the relation before enumeration, yielding similar benefits to Chain-of-Thought (CoT) prompting. Conversely, **CtE** sometimes lowers consistency due to higher levels of abstention caused by the more complex prompting strategy; for example, GPT-4.1-nano answers '`idk`' for classifying every $E_{4,1\setminus 3}$ instance, while **Base** does not include this step. Another cause for drops in performance relates to prompt misinterpretation; for example, GPT-5-nano often returns a relation for $E_{4,1\setminus 3}$ even during enumeration.

Oracle outperforms **Base** for all 18 models and exceeds **CtE** for 15 models, showing that relation information can improve answer-set consistency, especially when the correct relation is provided directly. When the model must infer the relation itself, as in **CtE**, performance often improves but can still be affected by abstention or prompt-misinterpretation errors.

An additional analysis in Appendix C quantifies output-format violations in saved responses. These are concentrated in Llama-3.1-8B, affecting 82.67% of its responses versus 1.45% for the remaining models. Excluding affected relations yields only modest consistency gains, indicating that substantial inconsistency persists beyond detectable format violations.

Hypothesis testing. Appendix J, reports statistical significance of our results; here, we summarize key findings. Hypothesis H_1 receives stronger support for **Oracle** than for **CtE**. As shown in Table 7, **Oracle** yields statistically significant improvements over **Base** for nearly all model-relation pairs. By contrast, **CtE** shows statistically significant gains for several relations and models, but the improvements are less uniform. Hypothesis H_2 is also supported, as the prompting strategies have a significant impact on both consistency and the relative ranking of models. We also observe significant positive correlations between LLM performance on the most challenging relations, particularly for $E_{4,1\setminus 3}$, and the models' average scores on the external benchmark (White et al., 2025) (see Table 8). In particular, $E_{4,1\setminus 3}$ exhibits strong positive correlations across both consistency measures.

5 Discussion

Our results show that answer-set inconsistency varies substantially across both models and set relations (RQ1). Larger LLMs perform better at recognizing set relations, but this trend is not uniform, with the best model being Gemini-2.5-pro (RQ2). For answer-set consistency, disjointness achieves the highest binary consistency rates, while containment and the more complex set-difference relation remain particularly challenging (RQ3). We further observe that inconsistency in equivalence judgments is largely driven by model stochasticity, whereas inconsistencies in other relations are primarily semantic in nature (RQ4). Finally, the prompting strategies suggest that making set-theoretic relations explicit to the model before enumeration can substantially reduce answer-set inconsistency (RQ5). Overall, the responses for LLMs to enumeration questions remain imperfect and exhibit inconsistency across repeated or related queries, leaving space for future work.

Limitations and Future Work. Answer-set (in)consistency could be extended to additional relations, including cardinality-based ones. Our experiments are limited to isolated, single-turn interactions; multi-turn dialogues may introduce or amplify inconsistencies. **ASCB** contains 600 English quadruples ($2{,}400$ questions) spanning static, factual domains. While the benchmark is manually curated and sufficiently large to support several statistically significant findings, future work could investigate automated methods for expanding its coverage. Another important direction is to study the relationship between accuracy and consistency using gold-standard answer sets, allowing inconsistencies caused by incorrect or missing knowledge to be distinguished from those that persist even when the underlying information is available and reliable. Further directions include prompting or fine-tuning for relation-aware reasoning (Ghosh et al., 2025; Calanzone et al., 2025), as well as hybrid approaches that map questions to structured representations and leverage external reasoning services.

AI USE STATEMENT

In this work, we used generative AI tools for text refinement, to assist code debugging, and for dataset creation (as described in the submission). We have not used generative AI tools for the development of the scientific ideas, methodology, experiments, analysis, or conclusions of this work. The research concept and scientific contributions were conceived and developed entirely by the authors. Generative AI tools were not used for any other tasks requiring disclosure, and the remaining disclosure categories are not applicable to this work.

REPRODUCIBILITY STATEMENT

We provide the complete ASCB benchmark, consisting of 600 question quadruples and $2{,}400$ questions, together with the expected set-theoretic relations used in our evaluation. The dataset is derived from LC-QUAD 2.0, QALD, QAWIKI, and an additional SYNTHETIC source, and was manually revised, filtered, and curated by three authors following the construction procedure described in Section 3.2 and Appendix . We release the dataset, source code, and experimental results through the anonymous repository (authors, 2026a) and Hugging Face dataset (authors, 2026b). The repository contains the resources required to reproduce the evaluation pipeline and compute the reported answer-set consistency, relation-classification, similarity, abstention, and statistical-significance measures. The prompts used for the different evaluation conditions are reported in Appendix F, while the dataset construction procedure is detailed in Appendix . All evaluated models are run using the lowest available temperature, and the paper reports the corresponding model configurations and evaluation protocol.

REFERENCES

Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases*. Addison-Wesley, 1995. ISBN 0-201-53771-0. URL http://webdam.inria.fr/Alice/.

David H. Ackley, Geoffrey E. Hinton, and Terrence J. Sejnowski. A learning algorithm for boltzmann machines. *Cognitive Science*, 9(1):147–169, 1985. ISSN 0364-0213. doi: 10.1016/S0364-0213(85)80012-4.

Dean Allemang and Juan Sequeda. Increasing the Accuracy of LLM Question-Answering Systems with Ontologies. In Gianluca Demartini, Katja Hose, Maribel Acosta, Matteo Palmonari, Gong Cheng, Hala Skaf-Molli, Nicolas Ferranti, Daniel Hernández, and Aidan Hogan (eds.), *The Semantic Web – ISWC 2024*, pp. 324–339, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-77847-6. doi: 10.1007/978-3-031-77847-6_18.

Berk Atıl, Sarp Aykent, Alexa Chittams, Lisheng Fu, Rebecca J. Passonneau, Evan Radcliffe, Guru Rajan Rajagopal, Adam Sloan, Tomasz Tudrej, Ferhan Ture, Zhe Wu, Lixinyu Xu, and Breck Baldwin. Non-Determinism of "Deterministic" LLM System Settings in Hosted Environments. In Mousumi Akter, Tahiya Chowdhury, Steffen Eger, Christoph Leiter, Juri Opitz, and Erion Çano (eds.), *Proceedings of the 5th Workshop on Evaluation and Comparison of NLP Systems*, pp. 135–148, Mumbai, India, December 2025. Association for Computational Linguistics. ISBN 979-8-89176-305-0. URL https://aclanthology.org/2025.eval4nlp-1.12/.

Anonymous authors. Answer set consistency of LLMs for question answering, 2026a. URL https://anonymous.4open.science/r/Answer-Set-Consistency-of-LLMs-For-QA-5FE6.

Anonymous authors. Answer set consistency benchmark, 2026b. URL https://huggingface.co/datasets/anonymous2026nips/llm-answer-set-qa.

Aayam Bansal and Ishaan Gangwani. Logical consistency under pressure: Probing and repairing cross-query contradictions in LLMs. In *ICLR 2026 Workshop on Logical Reasoning of Large Language Models*, 2026. URL https://openreview.net/forum?id=cz8eWlcZVF.

Diego Calanzone, Stefano Teso, and Antonio Vergari. Logically Consistent Language Models via Neuro-Symbolic Integration. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=7PGluppo4k.

Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. LM vs LM: Detecting factual errors via cross examination. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pp. 12621–12640. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.778.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FLASHATTENTION: fast and memory-efficient exact attention with IO-awareness. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088. doi: 10.5555/3600270.3601459.

Junhua Ding, Huyen Nguyen, and Haihua Chen. Evaluation of Question-Answering Based Text Summarization using LLM Invited Paper. In *2024 IEEE International Conference on Artificial Intelligence Testing (AITest)*, pp. 142–149, 2024. doi: 10.1109/AITest62860.2024.00025.

Mohnish Dubey, Debayan Banerjee, Abdelrahman Abdelkawi, and Jens Lehmann. LC-QuAD 2.0: A Large Dataset for Complex Question Answering over Wikidata and DBpedia. In Chiara Ghidini, Olaf Hartig, Maria Maleshkova, Vojtěch Svátek, Isabel Cruz, Aidan Hogan, Jie Song, Maxime Lefrançois, and Fabien Gandon (eds.), *The Semantic Web – ISWC 2019*, pp. 69–78, Cham, 2019. Springer International Publishing. ISBN 978-3-030-30796-7. doi: 10.1007/978-3-030-30796-7_5.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. Measuring and Improving Consistency in Pretrained Language Models. *Transactions of the Association for Computational Linguistics*, 9:1012–1031, 2021. doi: 10.1162/tacl_a_00410. URL https://aclanthology.org/2021.tacl-1.60/.

Bishwamittra Ghosh, Sarah Hasan, Naheed Anjum Arafat, and Arijit Khan. Logical Consistency of Large Language Models in Fact-Checking. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=SimlDuN0YT.

Aidan Hogan, Xin Luna Dong, Denny Vrandečić, and Gerhard Weikum. Large Language Models, Knowledge Graphs and Search Engines: A Crossroads for Answering Users' Questions. *arXiv preprint arXiv:2501.06699*, 2025. doi: 10.48550/arXiv.2501.06699.

Hasan Iqbal, Yuxia Wang, Minghan Wang, Georgi Nenkov Georgiev, Jiahui Geng, Iryna Gurevych, and Preslav Nakov. OpenFactCheck: A Unified Framework for Factuality Evaluation of LLMs. In Delia Irazu Hernandez Farias, Tom Hope, and Manling Li (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 219–229, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-demo.23.

Myeongjun Jang, Deuk Sin Kwon, and Thomas Lukasiewicz. BECEL: Benchmark for Consistency Evaluation of Language Models. In Nicoletta Calzolari, Chu-Ren Huang, Hansaem Kim, James Pustejovsky, Leo Wanner, Key-Sun Choi, Pum-Mo Ryu, Hsin-Hsi Chen, Lucia Donatelli, Heng Ji, Sadao Kurohashi, Patrizia Paggio, Nianwen Xue, Seokhwan Kim, Younggyun Hahm, Zhong He, Tony Kyungil Lee, Enrico Santus, Francis Bond, and Seung-Hoon Na (eds.), *Proceedings of the 29th International Conference on Computational Linguistics*, pp. 3680–3696, Gyeongju, Republic of Korea, October 2022. International Committee on Computational Linguistics. URL https://aclanthology.org/2022.coling-1.324/.

Peter A Lachenbruch. McNemar test. *Wiley StatsRef: Statistics Reference Online*, 2014. doi: 10.1002/9781118445112.stat04876.

Yanggyu Lee and Jihie Kim. Evaluating consistencies in LLM responses through a semantic clustering of question answering. *arXiv preprint arXiv:2410.15440*, 2024. doi: 10.48550/arXiv.2410.15440.

Lujun Li, Lama Sleem, Niccolo' Gentile, Geoffrey Nichil, and Radu State. Exploring the Impact of Temperature on Large Language Models: Hot or Cold? *Procedia Computer Science*, 264: 242–251, 2025. ISSN 1877-0509. doi: 10.1016/j.procs.2025.07.135. URL https://www.sciencedirect.com/science/article/pii/S1877050925021854. International Neural Network Society Workshop on Deep Learning Innovations and Applications 2025.

Nianqi Li, Jingping Liu, Sihang Jiang, Haiyun Jiang, Yanghua Xiao, Jiaqing Liang, Zujie Liang, Feng Wei, Jinglei Chen, Zhenghong Hao, and Bing Han. CR-LLM: A Dataset and Optimization for Concept Reasoning of Large Language Models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 13737–13747, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.815. URL https://aclanthology.org/2024.findings-acl.815/.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024a. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Shicheng Liu, Sina Semnani, Harold Triedman, Jialiang Xu, Isaac Dan Zhao, and Monica Lam. SPINACH: SPARQL-Based Information Navigation for Challenging Real-World Questions. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 15977–16001, Miami, Florida, USA, November 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.938. URL https://aclanthology.org/2024.findings-emnlp.938/.

Yinhong Liu, Zhijiang Guo, Tianya Liang, Ehsan Shareghi, Ivan Vulić, and Nigel Collier. Aligning with Logic: Measuring, Evaluating and Improving Logical Preference Consistency in Large Language Models. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=V61nluxFlR.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8086–8098, 2022. URL https://aclanthology.org/2022.acl-long.556.pdf.

Jiatong Ma, Linmei Hu, Rang Li, and Wenbo Fu. LoCal: Logical and Causal Fact-Checking with LLM-Based Multi-Agents. In *Proceedings of the ACM on Web Conference 2025*, WWW '25, pp. 1614–1625, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400712746. doi: 10.1145/3696410.3714748.

Saeed Masoudnia and Reza Ebrahimpour. Mixture of experts: a literature survey. *Artificial Intelligence Review*, 42(2):275–293, 2014. doi: 10.1007/s10462-012-9338-y.

Alberto Moya Loustaunau and Aidan Hogan. QAWiki: A knowledge graph question answering & SPARQL query generation dataset for wikidata. In *Wikidata Workshop, colocated with ISWC 2025*, 2025. URL https://ceur-ws.org/Vol-4108/paper12.pdf.

Nishanth Sridhar Nakshatri, Shamik Roy, Manoj Ghuhan Arivazhagan, Hanhan Zhou, Vinayshekhar Bannihatti Kumar, and Rashmi Gangadharaiah. When Facts Change: Probing LLMs on Evolving Knowledge with evolveQA. *arXiv preprint arXiv:2510.19172*, 2025. doi: 10.48550/arXiv.2510.19172.

Matthew Renze. The Effect of Sampling Temperature on Problem Solving in Large Language Models, November 2024. URL https://aclanthology.org/2024.findings-emnlp.432/.

Yash Saxena, Sarthak Chopra, and Arunendra Mani Tripathi. Evaluating Consistency and Reasoning Capabilities of Large Language Models. In *2024 Second International Conference on Data Science and Information System (ICDSIS)*, pp. 1–5, 2024. doi: 10.1109/ICDSIS61070.2024.10594233.

Aryan Singhal, Thomas Law, Coby Kassner, Ayushman Gupta, Evan Duan, Aviral Damle, and Ryan Luo Li. Multilingual Fact-Checking using LLMs. In Daryna Dementieva, Oana Ignat, Zhijing Jin, Rada Mihalcea, Giorgio Piatti, Joel Tetreault, Steven Wilson, and Jieyu Zhao (eds.), *Proceedings of the Third Workshop on NLP for Positive Impact*, pp. 13–31, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.nlp4pi-1.2. URL https://aclanthology.org/2024.nlp4pi-1.2/.

Yuan Sui, Yufei He, Zifeng Ding, and Bryan Hooi. Can Knowledge Graphs Make Large Language Models More Trustworthy? An Empirical Study Over Open-ended Question Answering. *arXiv preprint arXiv:2410.08085*, 2024.

Yiming Tan, Dehai Min, Yu Li, Wenbo Li, Nan Hu, Yongrui Chen, and Guilin Qi. Can ChatGPT Replace Traditional KBQA Models? An In-Depth Analysis of the Question Answering Performance of the GPT LLM Family. In Terry R. Payne, Valentina Presutti, Guilin Qi, María Poveda-Villalón, Giorgos Stoilos, Laura Hollink, Zoi Kaoudi, Gong Cheng, and Juanzi Li (eds.), *The Semantic Web – ISWC 2023*, pp. 348–367, Cham, 2023. Springer Nature Switzerland. ISBN 978-3-031-47240-4. doi: 10.1007/978-3-031-47240-4_19.

Ricardo Usbeck, Xi Yan, Aleksandr Perevalov, Longquan Jiang, Julius Schulz, Angelie Kraft, Cedric Möller, Junbo Huang, Jan Reineke, Axel-Cyrille Ngonga Ngomo, Muhammad Saleem, and Andreas Both. QALD-10 – the 10th challenge on question answering over linked data: Shifting from DBpedia to Wikidata as a KG for KGQA. *Semantic Web*, 15(6):2193–2207, 2024. doi: 10.3233/SW-233471.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964. doi: 10.5555/3295222.3295349.

Denny Vrandecic and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, 2014. doi: 10.1145/2629489. URL https://doi.org/10.1145/2629489.

Cunxiang Wang, Xiaoze Liu, Yuanhao Yue, Qipeng Guo, Xiangkun Hu, Xiangru Tang, Tianhang Zhang, Cheng Jiayang, Yunzhi Yao, Xuming Hu, Zehan Qi, Wenyang Gao, Yidong Wang, Linyi Yang, Jindong Wang, Xing Xie, Zheng Zhang, and Yue Zhang. Survey on Factuality in Large Language Models. *ACM Comput. Surv.*, 58(1), 2025. ISSN 0360-0300. doi: 10.1145/3742420. URL https://doi.org/10.1145/3742420.

Julian Junyan Wang and Victor Xiaoqi Wang. Assessing consistency and reproducibility in the outputs of large language models: Evidence across diverse finance and accounting tasks. *arXiv preprint arXiv:2503.16974*, 2025. doi: 10.2139/ssrn.5189069.

Anjiang Wei, Jiannan Cao, Ran Li, Hongyu Chen, Yuhui Zhang, Ziheng Wang, Yuan Liu, Thiago S. F. X. Teixeira, Diyi Yang, Ke Wang, and Alex Aiken. EquiBench: Benchmarking Large Language Models' Reasoning about Program Semantics via Equivalence Checking. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 33868–33881, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1718. URL https://aclanthology.org/2025.emnlp-main.1718/.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf.

Colin White, Samuel Dooley, Manley Roberts, Arka Pal, Benjamin Feuer, Siddhartha Jain, Ravid Shwartz-Ziv, Neel Jain, Khalid Saifullah, Sreemanti Dey, Shubh-Agrawal, Sandeep Singh Sandha, Siddartha Venkat Naidu, Chinmay Hegde, Yann LeCun, Tom Goldstein, Willie Neiswanger, and Micah Goldblum. LiveBench: A Challenging, Contamination-Limited LLM Benchmark. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=sKYHBTAxVa.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. Adaptive Chameleon or Stubborn Sloth: Revealing the Behavior of Large Language Models in Knowledge Conflicts. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=auKAUJZMO6.

Rongwu Xu, Zehan Qi, Zhijiang Guo, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. Knowledge Conflicts for LLMs: A Survey. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8541–8565, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.486. URL https://aclanthology.org/2024.emnlp-main.486/.

Jiayi Yuan, Hao Li, Xinheng Ding, Wenya Xie, Yu-Jhe Li, Wentian Zhao, Kun Wan, Jing Shi, Xia Hu, and Zirui Liu. Understanding and Mitigating Numerical Sources of Nondeterminism in LLM Inference. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=Q3qAsZAEZw.

Fuheng Zhao, Jiayue Chen, Lawrence Lim, Ishtiyaque Ahmad, Divyakant Agrawal, and Amr El Abbadi. LLM-SQL-Solver: Can LLMs Determine SQL Equivalence? *arXiv preprint arXiv:2312.10321*, 2025. doi: 10.48550/arXiv.2312.10321.

Danna Zheng, Mirella Lapata, and Jeff Z Pan. How Reliable are LLMs as Knowledge Bases? Re-thinking Facutality and Consistency. *arXiv preprint arXiv:2407.13578*, 2024. doi: 10.48550/arXiv.2407.13578.

Wenjie Zhou and Xiangyu Duan. Exploring and Improving Consistency in Large Language Models for Multiple-Choice Question Assessment. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–9, 2024. doi: 10.1109/IJCNN60899.2024.10650668.

Qinrui Zhu, Derui Lyu, Xi Fan, Xiangyu Wang, Qiang Tu, Yibin Zhan, and Huanhuan Chen. Multi-Model Consistency for LLMs' Evaluation. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2024. doi: 10.1109/IJCNN60899.2024.10651158.

A EXCLUSION OF 'IDK' RESPONSES

Table 4 reports results after excluding 'idk' responses from the evaluation and considering only non-'idk' answers. In contrast, the main-text results treat an 'idk' response generated by the model as incorrect, which penalizes models that abstain too often, preventing them from achieving artificially high consistency by repeatedly returning 'idk', which would make their outputs trivially consistent.

Importantly, however, the filtering in Table 4 is applied at the level of individual questions. Thus, if the model answers 'idk' for Q_1 but provides a non-'idk' answer for Q_3, only Q_1 is removed from the evaluation. As a result, metrics involving relations such as $N^{\text{CON}}_{3,1}$ may be inflated: removing one side of a relation can create a form of trivial consistency among the remaining answers.

Compared with Table 3, this table shows higher consistency values for many models because abstention are no longer penalized. This effect is strongest for model with large %IDK values, such as GPT-4o, GPT-5-mini, GPT-5, and GPT-5-nano. For instance, GPT-5-mini's **Base** CON scores rises from 22.20 in Table 3 to 55.27 in Table 4, while GPT-5's **Base** CON rises from 31.67 to 54.53. This suggest that part of the low performance in Table 3 was due to abstention of these models.

B NO-'IDK' ABLATION

To further analyze the behavior of LLMs when producing idk responses, we conduct an additional ablation study on 10 LLMs (GPT-4.1, GPT-4.1-mini, GPT-4.1-nano, GPT-4o, GPT-5-mini, GPT-5-nano, Gemini-2.5-Flash, Gemini-2.5-Pro, GPT-oss-20B, and Grok-3-mini), using the same prompting setup (see Appendix F) but removing the idk option in the prompt.

Macro-averaged results across the evaluated LLMs and the five CON predicates show that the difference between the no-idk and idk-enabled settings is $+2.36$ percentage points for **Base**, $+15.9$ points for **CtE**, and $+3.18$ points for **Oracle**, corresponding to an overall increase of $+7.1$ points. At the aggregate level, **CtE** and **Oracle** continue to outperform **Base**, preserving the main qualitative pattern observed under the idk-enabled prompt.

Manual inspection of the generated answers, however, reveals that removing the standardized idk option does not eliminate abstention, but instead changes how it is expressed. Examples include "none", "Return no answer", "As of my last knowledge update", and question-specific statements such as "I do not have the exact current rankings of the 10 richest people in the world". Unlike standardized idk and no-answer tokens, these open-ended abstentions cannot be exhaustively identified through deterministic string matching. Moreover, when treated as ordinary answer elements, they may mechanically satisfy some set relations. For example, identical "none" responses to Q_1 and Q_2 satisfy $E^{\text{CON}}_{1,2}$, whereas different abstention phrases for Q_3 and Q_4 may satisfy $D^{\text{CON}}_{3,4}$ because the resulting singleton sets are disjoint.

We therefore do not interpret the higher scores obtained in the no-idk setting as direct evidence of improved reasoning. Rather, the ablation shows that the availability of an explicit idk option affects model behavior, and that removing it makes abstention less explicit and less reliably measurable. This result further motivates the use of a standardized idk option, which makes abstention directly observable and enables us either to assign a score of zero to consistency predicates affected by abstention (Table 3) or to exclude such answers from the evaluation (Table 4).

C OUTPUT-FORMAT VIOLATIONS

To distinguish answer-set inconsistency from abstention and output-format noncompliance, we conducted a conservative audit of the saved responses under the **Base** strategy. The audit detects explanations, line breaks, Markdown, prompt echoes, noncanonical abstentions, and null tokens mixed with enumerated answers. Detectable violations occurred in 2,573 of 43,120 responses (5.97%), with substantial variation across models: Llama-3.1-8B exhibited a violation rate of 82.67% (1,984/2,400), compared with 1.45% across the remaining models (589/40,720). We then recomputed consistency under two nested exclusions: first removing each relation instance containing an idk or empty response required by its predicate, and then additionally removing instances involving a detectable format violation. Pooling eligible relation instances across models, the additional format

Table 4: Per-relation consistency (0-100 scale) for each model and strategy (Str). When LLMs responds with "IDK" the question is not included in the metric evaluation.

ID	Model	Str	$E^{\text{CON}}_{1,*}$	$E^{\text{SIM}}_{1,*}$	$E^{\text{CON}}_{1,2}$	$N^{\text{CON}}_{3,1}$	$N^{\text{CON}}_{4,1}$	$D^{\text{CON}}_{3,4}$	$E^{\text{CON}}_{4,1\setminus 3}$	CON	$E^{\text{SIM}}_{1,2}$	$D^{\text{SIM}}_{3,4}$	$E^{\text{SIM}}_{4,1\setminus 3}$	%IDK
A	Llama-3.1-8b	Base	3.60	22.22	4.17	12.00	7.83	69.17	0.00	18.63	25.87	8.17	10.37	0.75
		CtE	–"–	–"–	16.33	9.83	33.83	68.17	5.00	26.63	56.03	9.74	38.99	**0.08**
		Ora.	–"–	–"–	12.67	15.50	12.33	61.83	1.83	20.83	45.49	11.26	24.14	0.21
B	GPT-oss-20b	Base	22.71	44.44	22.50	38.00	32.00	82.50	9.50	36.90	41.55	7.87	26.99	14.84
		CtE	–"–	–"–	67.17	39.33	73.17	84.67	38.00	60.47	79.39	10.73	60.99	15.75
		Ora.	–"–	–"–	58.33	75.67	92.33	95.33	62.00	76.73	76.50	3.97	78.64	11.92
C	gpt-4.1-nano	Base	56.89	66.31	29.00	42.17	45.33	65.33	4.50	37.27	51.00	15.59	22.20	13.79
		CtE	–"–	–"–	**97.33**	67.33	56.67	38.17	0.00	51.90	97.57	61.49	0.17	79.83
		Ora.	–"–	–"–	54.67	55.50	60.17	61.83	21.50	50.73	62.37	22.57	35.09	18.58
D	Mistral-small:24b	Base	36.25	52.48	33.90	40.51	45.08	50.85	0.68	34.20	44.70	34.47	21.53	38.69
		CtE	–"–	–"–	84.50	44.67	80.67	79.83	42.33	66.40	92.57	17.74	63.71	11.42
		Ora.	–"–	–"–	72.00	69.00	90.00	95.83	50.17	75.40	82.96	1.48	67.82	15.33
E	Llama-3.1-70b	Base	20.33	48.82	17.00	27.33	23.83	70.33	2.33	28.16	41.24	10.45	19.19	2.75
		CtE	–"–	–"–	80.33	37.83	86.33	93.50	56.17	70.83	90.71	2.95	81.20	2.37
		Ora.	–"–	–"–	61.00	70.17	82.17	93.50	50.67	71.50	75.29	3.37	76.35	4.54
F	Gemini-2.0-flash	Base	48.03	70.08	32.67	41.50	36.00	64.50	5.50	36.03	60.08	11.38	25.44	3.00
		CtE	–"–	–"–	88.17	53.00	94.50	64.33	53.83	70.77	93.84	27.82	62.68	22.46
		Ora.	–"–	–"–	79.00	84.17	92.00	96.67	73.83	85.13	82.33	**1.37**	84.15	2.96
G	gpt-4.1-mini	Base	28.02	53.84	17.50	44.33	23.50	75.33	3.83	32.90	35.72	7.64	16.50	8.21
		CtE	–"–	–"–	89.83	49.67	94.83	77.17	64.00	75.10	95.73	21.83	72.41	18.54
		Ora.	–"–	–"–	77.50	86.50	96.33	89.67	68.83	83.77	88.81	6.10	79.53	8.63
H	GPT-4o	Base	48.98	52.09	33.00	35.83	31.83	19.50	1.50	24.33	34.65	78.93	4.78	71.08
		CtE	–"–	–"–	**97.33**	52.33	98.33	36.33	30.67	63.00	58.67	62.79	35.08	59.83
		Ora.	–"–	–"–	85.33	85.17	94.83	86.50	67.83	83.93	73.43	12.09	58.79	32.33
I	gpt-4.1	Base	28.05	58.83	31.00	51.50	40.83	63.83	11.50	39.73	61.04	10.77	37.14	3.75
		CtE	–"–	–"–	91.83	57.50	97.50	87.00	74.83	81.73	97.54	11.74	**84.84**	8.16
		Ora.	–"–	–"–	82.50	91.00	94.50	95.33	74.83	87.63	92.07	3.09	83.64	6.42
J	Grok-3-mini	Base	37.67	63.09	29.17	52.33	41.50	88.83	21.00	46.57	54.84	4.00	42.03	8.54
		CtE	–"–	–"–	90.83	57.67	90.00	86.67	66.83	78.40	96.39	12.77	78.69	13.66
		Ora.	–"–	–"–	88.17	90.83	**98.50**	95.50	81.00	90.80	93.63	4.00	79.16	13.25
K	DeepSeek-V3.1	Base	38.81	60.82	32.50	47.17	40.17	58.83	6.83	37.10	57.03	15.15	25.99	12.67
		CtE	–"–	–"–	94.50	74.67	91.17	67.33	54.17	76.37	97.22	31.01	62.74	27.21
		Ora.	–"–	–"–	81.67	85.67	95.00	94.67	71.00	85.60	91.17	3.96	79.80	9.79
L	Gemini-2.5-flash	Base	37.69	64.85	33.50	49.50	44.00	88.33	18.00	46.67	61.01	2.22	33.56	8.58
		CtE	–"–	–"–	89.83	57.67	96.83	90.67	84.83	83.97	93.83	8.82	83.92	10.00
		Ora.	–"–	–"–	89.17	89.00	95.33	95.67	**85.50**	**90.93**	92.56	4.08	83.49	7.83
M	GPT-5-nano	Base	33.48	42.09	59.00	63.17	63.17	64.67	16.00	53.20	72.28	33.29	35.76	43.50
		CtE	–"–	–"–	77.83	7.67	76.83	19.50	0.50	36.47	78.16	80.39	7.82	7.17
		Ora.	–"–	–"–	84.17	81.83	97.17	77.83	53.33	78.87	89.65	21.40	58.45	41.12
N	DeepSeek-R	Base	29.95	51.86	18.67	40.17	30.17	82.00	10.83	36.37	46.10	3.96	31.15	4.21
		CtE	–"–	–"–	78.50	62.17	85.33	69.67	51.50	69.43	82.40	29.66	56.37	23.75
		Ora.	–"–	–"–	68.33	74.17	86.33	93.00	58.33	76.03	82.21	5.37	79.55	5.92
O	Gemini-2.5-pro	Base	32.14	59.69	30.17	49.00	33.33	83.50	17.50	42.70	63.02	2.88	40.93	3.29
		CtE	–"–	–"–	85.33	58.33	88.00	93.00	71.33	79.20	89.78	6.22	79.02	11.08
		Ora.	–"–	–"–	77.33	73.17	92.17	**97.83**	74.83	83.07	81.97	2.06	80.89	8.25
P	GPT-5-mini	Base	**60.77**	72.54	61.83	72.33	75.50	51.17	15.50	55.27	70.04	46.74	32.82	55.12
		CtE	–"–	–"–	88.00	63.17	75.00	67.33	34.00	65.50	92.58	32.67	44.64	39.71
		Ora.	–"–	–"–	86.83	88.83	95.00	73.67	54.50	79.77	91.04	26.17	55.03	49.38
Q	GPT-o3	Base	31.42	58.92	27.12	45.93	34.75	89.49	16.61	42.78	52.91	4.40	40.75	9.83
		CtE	–"–	–"–	82.00	52.00	91.33	92.50	74.67	78.50	93.24	6.74	83.97	9.62
		Ora.	–"–	–"–	73.67	90.00	94.50	97.33	77.17	86.53	82.18	2.44	80.82	13.38
R	GPT-5	Base	58.59	**76.89**	54.67	64.17	61.50	73.33	19.00	54.53	68.76	23.04	43.75	37.50
		CtE	–"–	–"–	92.33	67.67	96.50	70.67	59.83	77.40	**96.63**	29.20	66.83	30.29
		Ora.	–"–	–"–	90.67	**93.67**	98.17	79.00	65.33	85.37	96.11	21.00	69.17	34.54

filtering increased consistency by 2.24 percentage points for $E^{\text{CON}}_{1,2}$, 3.21 for $N^{\text{CON}}_{3,1}$, 2.71 for $N^{\text{CON}}_{4,1}$, 1.53 for $E^{\text{CON}}_{4,1\setminus 3}$, and 0.45 for $D^{\text{CON}}_{3,4}$. The resulting consistency rates were 29.26%, 46.52%, 35.64%, 14.07%, and 73.98%, respectively. Thus, detectable format violations are concentrated in one model, and excluding affected instances yields modest pooled improvements. Substantial answer-set

inconsistency persists among non-abstaining responses without detectable format violations. These comparisons describe consistency on progressively filtered subsets, rather than the causal effect of correcting output formatting.

D Entity disambiguation with Wikidata

In the prompt used with the models, we explicitly instructed the LLMs to normalize their answers accordingly F, for example by expanding acronyms (United States of America instead of U.S.A) or using Arabic numerals (World War 2 instead of World War II). Lexical variants were therefore considered inconsistent in our evaluation, as they violated these instructions. To quantify this terminology issue, we parsed all enumerated answers returned by the LLMs and used Wikidata aliases API for entity disambiguation, which list, for instance, Kingdom of Spain as an alias of Spain. These results are reported in Table 5. Gemini-2.5-pro was the most affected model, with 136 lexical inconsistencies out of $2,400$ questions (5.6%), while Mistral-small:24b was the least affected, with 88 cases (3.6%).

Table 5: Lexical inconsistencies by model. Counts are reported with the corresponding percentage of affected answers.

ID	Model	Count	Rate (%)
A	Llama-3.1-8b	105	4.3
B	GPT-oss-20b	114	4.7
C	GPT-4.1-nano	98	4.2
D	Mistral-small-24b	88	3.6
E	Llama-3.1-70b	117	4.8
F	Gemini-2.0-flash	110	4.5
G	GPT-4.1-mini	111	4.6
H	GPT-4o	101	4.2
I	GPT-4.1	116	4.8
J	Grok-3-mini	110	4.5
K	DeepSeek-V3.1	122	5.0
L	Gemini-2.5-flash	106	4.4
M	GPT-5-nano	91	3.7
N	DeepSeek-Reasoner	130	5.4
O	Gemini-2.5-pro	136	5.6
P	GPT-5-mini	98	4.2
Q	GPT-o3	131	5.4
R	GPT-5	108	4.5

E Question Wording and Negation

Although $N_{3,1}$ and $N_{4,1}$ evaluate the same abstract containment relation, under **Base**, $N_{4,1}^{\mathrm{CON}} < N_{3,1}^{\mathrm{CON}}$ for every model. This ordering is not consistently observed under **CtE** or **Oracle**. Because Q_4 expresses a complementary restriction, we investigated whether explicit negation or exclusion markers are associated with lower consistency. Using case-insensitive matching for "not" "never", "excluding", "except", "without", "outside", "no longer", "n't", "non-", "other than", and "aside from", we identified markers in 399 of 600 Q_4 questions (66.50%), compared with 38 Q_3 questions (6.33%). Under **Base**, with relevant `idk`/empty responses scored as inconsistent, pooled $N_{4,1}^{\mathrm{CON}}$ was 22.90% for marked Q_4 questions and 25.65% for unmarked questions, a difference of -2.75 percentage points. This difference was not statistically significant ($p = 0.159$; two-sided Welch t-test, aggregated across models within each question group). For Q_1, Q_2, equivalence consistency was 6.21% in the 17 groups with a marker in either question, compared with 21.31% in the remaining 583 groups ($p = 0.00077$, using the same test). However, 14 of those 17 groups contained markers in both questions, leaving only three pairs with differing marker presence. Overall, explicit markers are more frequent in complementary questions, but these observational comparisons do not establish negation as the cause of the containment asymmetry. Question content, answer-set composition, and wording remain possible contributors.

F PROMPTS AND PROCESSES USED WITH LLMS

This section presents the three prompting strategies employed to evaluate the LLMs. The placeholder question denotes the text segment that is replaced with a question from the dataset.

F.1 *Zero-shot*

```
{question}
If you can't answer, return 'idk'.
If the question has no answer, return 'no answer'.
In the response, do not use abbreviations or acronyms, but spell out the full terms, i.e.  "United States of America" instead of "USA".
If the response contains numbers or digits, use Arabic numerals. For example, if the answer contains Star Wars V, indicate it with Star Wars 5. Do not use Roman numerals (such as V) or text (such as five).
Please, Return me an exhaustive list separated by the symbol '|' don't add any other text.
```

F.2 *Classification & Question*

```
You are given two questions, q1 and q2.
Your task is to determine the logical relationship between their respective sets of correct answers
Choose only one of the following relations:
- Equivalence: The answer sets of q1 and q2 are exactly the same.
- Contains: All answers to q2 are also answers to q1, but q1 includes additional answers.
- ContainedBy: All answers to q1 are also answers to q2, but q2 includes additional answers.
- Overlap: q1 and q2 share some, but not all, answers. Neither fully contains the other.
- Disjoint: q1 and q2 have no answers in common.
- Unknown: The relation between the answer sets cannot be confidently determined based on the given questions.
Here are the two questions:
q1: {q1}
q2: {q2}
Return **only** the name of the most appropriate relation from the list above.
Do **not** provide any explanation or commentary.
```

```
You are given three questions: q1, q2, and q3.
Each question is associated with a set of answers. Your task is to identify the logical relation between the concepts of questions based on their answer sets. Compare the relationship of concept between the following two sets, s1 and s2:
  - s1: the set of all answers for q1 that are not answers for q2
  - s2: the set of answers for q3.

These are the three questions:
q1: {q1}
q2: {q2}
q3: {q3}
For each comparison, use **one of the following labels**:
- Equivalence
- Contains
- ContainedBy
- Disjoint
- Overlap
- Unknown
Return only the exactly relation label.
Do **not** include any explanation or extra text.
```

F.3 *Oracle*

Below is the oracle prompt used for the logical equivalence relation. For the other two other logical relations tested, the prompt is adapted by replacing the `true_logical_relation` placeholder and modifying certain parts of the sentence to ensure coherence.

```
Pay attention, the questions I asked you before are {
    true_logical_relation}, but you returned me different values.
In the response, do not use abbreviations or acronyms, but spell out the
    full terms, i.e. "United States of America" instead of "USA".
If the response contains numbers or digits, use Arabic numerals. For
    example, if the answer contains Star Wars V, indicate it with Star
    Wars 5. Do not use Roman numerals (such as V) or text (such as five).
Please, Return me an exhaustive list separated by the symbol '|' don't
    add any other text.
```

F.4 Evaluation pipeline

After submitting the questions contained in the ASCB using the prompts illustrated previously, with each treated as an independent question, the responses are stored in JSON files where the key corresponds to the question ID and the value is a list of elements representing the set of answers generated by the LLM for that particular question. Once the benchmark has been executed across all models and for all three task types (**Base**, **CtE**, and **Oracle**), all JSON files are provided as input to an evaluation pipeline. This pipeline produces TSV files, one for each evaluated LLM, containing for every answer: the Jaccard similarity, consistency, and a boolean flag indicating whether the answer to the logical relation was satisfied. Finally, as the last step, starting from these task-, model-, and dataset-specific TSV files, four distinct summary TSV files (one for each dataset) are generated. These summary files report the following information: the type of logical relation evaluated, the LLM, the action, the consistency rates, the average non-empty response rate, the average Jaccard similarity, and the ratio of empty responses. All aforementioned TSV file are publicly available in the GitHub repository provided as supplementary material authors (2026a).

G Dataset construction

We describe now the dataset construction, consisting of our extraction and modification of questions from the base datasets, followed by a process to filter and identify suitable candidate questions.

G.1 Base datasets

To construct the question sets, we relied on four data sources: QALD (Liu et al., 2024b; Usbeck et al., 2024), a handcrafted question-answering dataset for Wikidata and DBpedia; LC-QuAD 2.0 (Dubey et al., 2019), a large-scale question-answering dataset providing SPARQL queries with corresponding answers from both Wikidata and DBpedia[3] ; and a Synthetic dataset, generated entirely through LLMs.

QALD and LC-QuAD 2.0. To extract questions from QALD, and LC-QuAD 2.0 datasets that conformed to our criteria, we employed a filtering step using an LLM-based pipeline, described in detail in the Section G.2. The resulting questions were reviewed by three authors to ensure adherence to the defined logical relationships and criteria; we remark that careful revisions and considerable adaptations were required by hand, with many candidate questions being discarded. The final filtered QALD and LC-QuAD 2.0 datasets each contain 150 distinct question quadruples (300 for both), for a total of 600 questions in each dataset ($1{,}200$ across both).

QAWiki. For the QAWiki dataset, many Q_1, Q_2, and Q_3 pairs were directly retrieved via SPARQL queries. However, only 54 Q_1–Q_3 pairs and 951 Q_1–Q_2 pairs were available. Therefore, for the 54 Q_1–Q_3 pairs, the corresponding Q_2 and Q_4 questions were manually constructed. Conversely, for the 951 Q_1–Q_2 pairs, Q_3 and Q_4 were manually derived. As with the other sources, this dataset

[3]DBpedia: `https://www.dbpedia.org/`

was manually curated to include 150 distinct question quadruples per the required relations and criteria, totaling 600 questions.

SYNTHETIC. The SYNTHETIC dataset was generated using GPT-4.1-2025-04-14, designed to produce 500 complete sets of Q_1, Q_2, Q_3, and Q_4 questions. Each set was subsequently reviewed independently by three of the article's authors to ensure accuracy, confirm compliance with the inclusion criteria, and remove any duplicate entries. This dataset was heavily curated and revised by hand to include 150 distinct question quadruples, totaling 600 questions. In many cases, the base question tuple generated by the LLM served only as inspiration for the final handcrafted question.

The prompt used to generate the pair of questions Q_1 and Q_2 is the following:

```
Generate {number_of_questions_to_generate} pairs of diverse questions about different topics, every pair of questions must be semantically equivalent.
The answer to every question that you formulate must be a list of values, not an ordered list, not a paragraph of text, not a boolean value, and not a single number.

This is an example of a possible pair of questions:
1. How many regions of France are there? | How many regions does France have?
Follow the following format to return the questions:
1. Question1 | Equivalent_Question1
Do not add any other kind of text except questions.'
```

Following the initial generation of the questions, three of the authors conducted a first round of revision to verify their correctness and semantic equivalence. Based on this refined pool of questions, we then instructed the LLM to generate, for each pair of equivalent questions (Q_1) and (Q_2), a third question (Q_3) whose answer set constitutes a subset of the answers to both (Q_1) and (Q_2). The prompt used for this task is reported below:

```
{dataset_of_question_pairs}
Starting from the provided dataset of question pairs, where each pair consists of two semantically equivalent questions, generate a third question whose answer is a subset of the answers to the original two questions.
The answer to the generated question must be a list of values (not an ordered list, not a descriptive paragraph, not a Boolean value, and not a single number).

Use the following output format, and provide only the third question:

1. Broader_Question_from_the_dataset | Subset_Question

For example, given the pair: "What countries are in the EU?" | "What countries are in the western EU?" the generated question would be the subset question.

Return only the formulated subset question and no additional text.
```

Once the third question was generated for each case, we manually reviewed each (Q_3) and constructed a corresponding fourth question (Q_4) to satisfy the disjointness relation, defined as ($Q_4 = Q_1 \setminus Q_3$). The final dataset comprises 600 question quadruples

G.2 QUESTION PIPELINE

The multi-agent pipeline designed to filter questions that do not satisfy our inclusion criteria was instructed to evaluate each input question against the defined conditions and, upon validation, to generate the corresponding Q_2, Q_3, and Q_4 questions. The LLM employed for this task was GPT-4.1-2025-04-14. The initial datasets consisted of 320 questions from QALD and 30,000 questions from LC-QUAD 2.0. The multi-agent workflow comprises four sequential stages:

- **Agent 1 (Validate Q1):** checks whether the input question Q_1 is well-formed and suitable; aborts if invalid.
- **Agent 2 (Generate Q2):** produces a companion question Q_2 consistent with the intent and constraints of Q_1.
- **Agent 3 (Generate Q3 & Q4):** creates Q_3 and Q_4 such that the quadruple (Q_1, Q_2, Q_3, Q_4) satisfies the desired logical relations (e.g., equivalence, containment, or disjointness).
- **Agent 4 (Validate All):** verifies that (Q_1, Q_2, Q_3, Q_4) meet the required logical and formatting rules, corrects minor issues, and outputs the final validated set.

The process aborts if any step fails.

Agent1:

```
Evaluate whether the following question meets all of the following
criteria for acceptable answer types:

1. Returns a limited number of distinct answers (between 2 to 50).
2. Does **not** return a binary answer (e.g., "yes" or "no").
3. Does **not** return a single specific value (e.g., a date, name,
or number).
4. Does **not** require multiple answer dimensions (e.g., combining "
what" and "where" in the same question).

Question: "{question}"

Answer only with "Yes" or "No".
```

Agent2:

```
You are a rephrasing expert. Generate question Q2 which means exactly
 the same thing as the original question but uses different syntax
wording.

Original Question: "{question}"

Format your response as:
Q2: ...
```

Agent3:

```
Given the original question below, generate Q3 and Q4 to ensure:
- Q3 and Q4 are objective,
- answer set of Q1 equal to union of answer set of Q3 and Q4,
- answer set of Q3 disjoint from Q4,
- answer set of Q3 and Q4 subset of Q1 (more restrictive).

Original Question: "{question}"

Format your response as:
Q3: ...
Q4: ...
```

Agent4:

```
You are reviewing four related questions.

Q1: {q1}
Q2: {q2}
Q3: {q3}
Q4: {q4}

Tasks:
1. Check if Q2 is equivalent to Q1.
2. Check if Q3 is a more restrictive version of Q1.
```

```
    3. Check if the union of Q3 and Q4 covers Q1 (Q1 = Q3 + Q4).
    4. Check all Q1~Q4 are objective questions.

    If everything is correct, answer only:
    "Yes"

    If not, return a corrected version in this format:
    Corrected Q1: ...
    Corrected Q2: ...
    Corrected Q3: ...
    Corrected Q4: ...
```

G.3 Dataset License and Attribution

ASCB was constructed using LC-QuAD 2.0, QALD,QAWiki, and a SYNTHETIC subset. Source questions were filtered, transformed, manually revised, paraphrased, and organized into relation-based quadruples. Some source-derived questions, in particular certain Q_1 questions and some QAWiki-derived Q_1-Q_3 pairs, may remain unchanged or only lightly modified. LC-QuAD 2.0 is released under GPL-3.0, QALD under CC BY 4.0, and QAWiki under CC0. Because **ASCB** includes material derived from LC-QuAD 2.0, the combined ASCB dataset is released under GPL-3.0. We retain attribution and license notices for all source datasets and indicate that derived questions may have been modified during construction.

H Visualizations

In this section, we visualize the main answer-set inconsistency results across models. These charts are based on results obtained considering wrong the '`idk`' answers, i.e., the results reported in Table 3 in the main text.

Figure 1 presents the Jaccard similarity $E^{\text{SIM}}_{1,2}$, $D^{\text{SIM}}_{3,4}$ and $E^{\text{SIM}}_{4,1\setminus 3}$. Figure 1a illustrates that for $E^{\text{SIM}}_{1,2}$ there is high consistency across models, particularly for **CtE** and **Oracle**, with Jaccard similarity values frequently exceeding 0.8. Figure 1b illustrates the consistency for $D^{\text{SIM}}_{3,4}$. In this setting, lower Jaccard similarity indicates better performance, as the answers are expected to be disjoint. The results show that **Oracle** achieves the best overall performance, while **Base** and **CtE** obtain similar results. This is mainly because, under **CtE**, models tend to exhibit increased confusion, leading to a higher abstention rate among LLMs and ultimately resulting in lower performance compared to the other strategies. Finally, Figure 1c shows that consistency for $E^{\text{SIM}}_{4,1\setminus 3}$ is low compared to $E^{\text{SIM}}_{1,2}$. Although adjusting the prompting strategy can improve performance, the gains are more consistent under the **Oracle** strategy. In contrast, under **CtE**, some models perform worse on $E^{\text{SIM}}_{4,1\setminus 3}$ than on $E^{\text{SIM}}_{1,2}$.

I Classification Task Result

Table 6 reports in detail the accuracy of the different models in identifying the different relationships. Across all models, the performance varies from $\sim$60–91% in terms of average accuracy, depending on the relation. Interestingly, the models perform best, on average, for the disjointness relation ($D_{3,4}$), with an average accuracy across models of $\sim$90.48%. The most difficult relationship to recognize for all models was $N_{4,1}$, with an average accuracy of $\sim$60.64%. A number of models achieve average accuracy above 90%, including some smaller models, with the best performing (taking ACC: the average across relations) being Gemini-2.5-pro, followed closely by GPT-5, Grok-3-mini, and other models, all of which achieve average accuracy across relations of $\sim$95%.

J Statistical Significance

The statistical significance test is performed using the results obtained by treating '`idk`' answers as incorrect, namely the results reported in Table 3 in the main text.

Table 7 presents the McNemar p-values used to assess whether the **CtE** and **Oracle** strategies significantly improve answer-set consistency over the **Base** strategy on the combined dataset. The

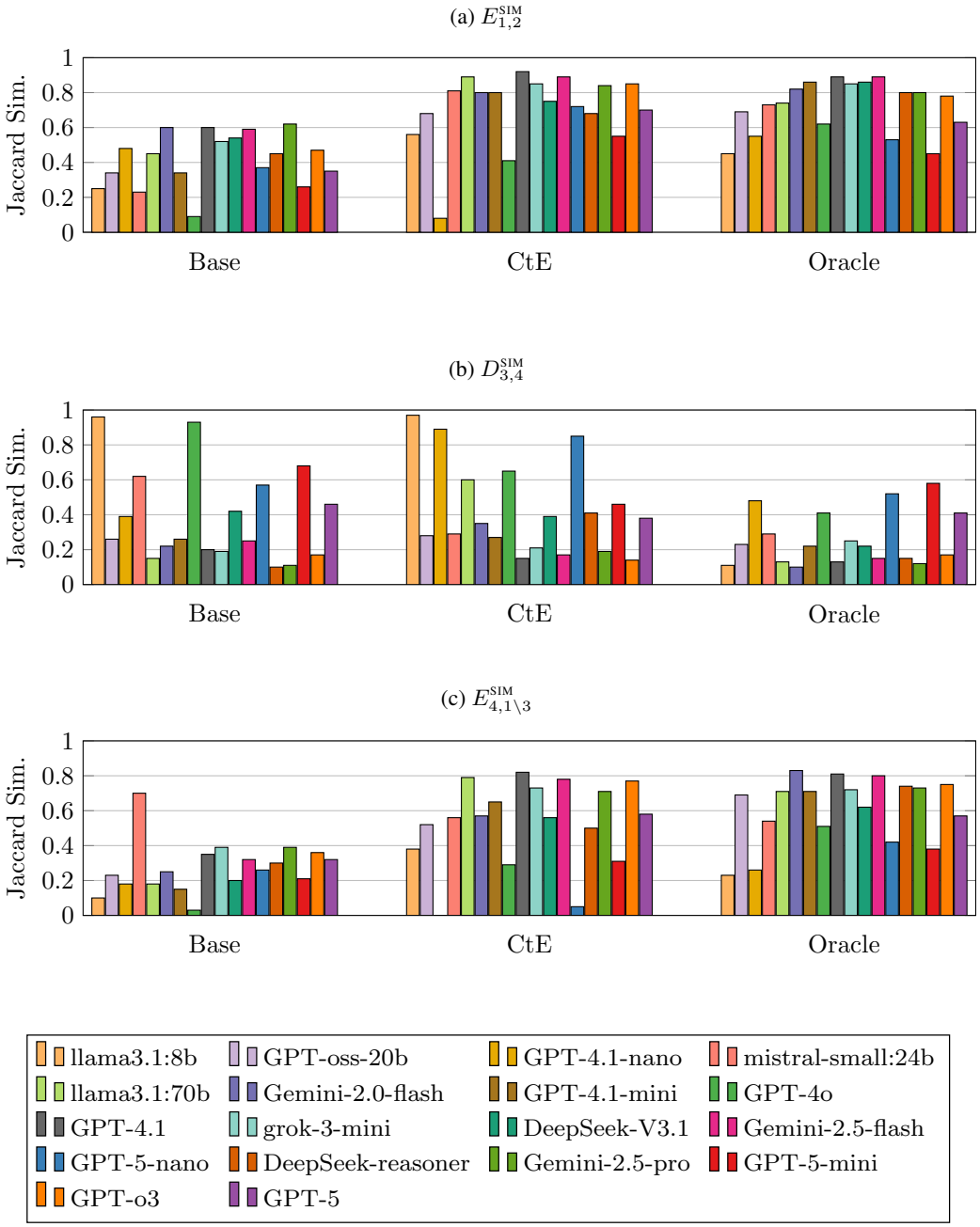

Figure 1: Visualization of answer-set (in)consistency across models

McNemar test is a paired, non-parametric test that considers only cases where the two strategies disagree. In our setting, each evaluated instance is binary: the expected relation either holds between the generated answer sets or it does not. We apply the test separately for each model and relation, using the same consistency outcomes reported in the main results.

The results show that relation-aware prompting often improves answer-set consistency, but the effect depends strongly on both the strategy and the relation. **CtE** yields significant improvements for all models on $E_{1,2}$ and for many models on $N_{4,1}$ and $E_{4,1\setminus 3}$. However, the gains are less uniform for $N_{3,1}$ and especially for $D_{3,4}$, where several models do not show statistically significant improvement.

Table 6: Accuracy of different LLMs on the relation classification task across relations. In bold, the model that achieves the highest accuracy for the relation(s). ACC indicates mean accuracy.

ID	Model	$E^{\text{ACC}}_{1,2}$	$N^{\text{ACC}}_{3,1}$	$N^{\text{ACC}}_{4,1}$	$D^{\text{ACC}}_{3,4}$	$E^{\text{ACC}}_{4,1\setminus 3}$	ACC
A	Llama-3.1-8b	5.83	11.33	11.50	19.17	2.83	10.13
B	GPT-oss-20b	93.83	92.33	96.33	91.33	94.00	93.56
C	GPT-4.1-nano	69.17	39.33	41.33	62.33	0.00	42.43
D	Mistral-small:24b	91.69	0.17	1.19	99.49	11.86	40.88
E	Llama-3.1-70b	**97.33**	7.67	27.00	99.17	98.50	65.93
F	Gemini-2.0-flash	90.83	79.17	21.33	95.17	80.33	73.37
G	GPT-4.1-mini	91.00	34.50	20.83	99.00	95.83	68.23
H	GPT-4o	94.50	31.67	12.17	99.33	65.33	60.60
I	GPT-4.1	94.50	**98.67**	69.67	**99.50**	**99.33**	92.33
J	Grok-3-mini	91.67	94.83	96.50	97.67	96.83	95.50
K	DeepSeek-V3.1	96.00	35.50	20.83	99.33	90.50	68.43
L	Gemini-2.5-flash	86.33	95.67	96.33	96.83	95.50	94.13
M	GPT-5-nano	91.83	85.00	91.50	88.50	91.17	89.60
N	DeepSeek-reasoner	88.83	97.00	96.67	96.33	94.33	94.63
O	Gemini-2.5-pro	91.33	97.67	94.50	98.50	96.17	**95.63**
P	GPT-5-mini	90.33	97.17	97.83	95.00	95.17	95.10
Q	GPT-o3	92.71	97.12	97.63	95.25	93.73	95.29
R	GPT-5	90.50	97.67	**98.33**	96.83	94.50	95.57
Average		86.01	66.25	60.64	**90.48**	77.55	76.19

This indicates that asking the model to classify the relation before enumeration can help, but only when the model is able to use the inferred relation reliably during answer generation.

The **Oracle** strategy provides much stronger support for H_1. When the correct relation is explicitly given to the model, improvements over **Base** are statistically significant for almost every model-relation pair. The only nonsignificant case in Table 7 is $D_{3,4}$ for GPT-4.1-nano. This suggests that explicit and correct relation information can substantially reduce answer-set inconsistency, although the extent of the improvement still depends on the relation.

To evaluate H_2, we examine whether external model performance correlates with answer-set consistency. Table 8 reports Pearson correlations between the external benchmark scores and the reasoning performance metrics. Models not included in the external benchmark are excluded. The strongest and statistically significant correlations are observed for the set-difference relation $E_{4,1\setminus 3}$: $E^{CON}_{4,1\setminus 3}$ has $r = 0.827$ with $p < 0.001$, and $E^{SIM}_{4,1\setminus 3}$ has $r = 0.70$ with $p < 0.004$. By contrast, the remaining metrics do not show statistically significant correlations at the $p < 0.05$ level. The correlations for $D^{CON}_{3,4}$ and $CON()$ are positive but only marginal, with $p = 0.05$ and $p = 0.06$, respectively.

These results support H_2 primarily for the most complex answer-set task. In particular, the strong correlations for $E_{4,1\setminus 3}$ suggest that external model strength is most predictive when the consistency task requires more complex set-theoretic reasoning, rather than for simpler binary relations such as equivalence or containment.

Table 8 shows significant positive correlations between $D^{\text{CON}}_{3,4}$, $E^{\text{CON}}_{4,1\setminus 3}$, and $D^{\text{SIM}}_{3,4}$ and the average score of the models on the external benchmark (White et al., 2025). Models not in this benchmark are excluded. In particular, $E_{4,1\setminus 3}$ exhibits strong positive correlations across both measures. Since $E_{4,1\setminus 3}$ is the most challenging relation to identify, this finding supports H_2, suggesting that consistency is more pronounced in the more complex answer-set tasks.

K Anwser-Set Contradictions

In the body of the paper, we have looked at answer-set inconsistencies with respect to gold-standard relations between questions. We further define *answer-set contradictions* as the case where – irrespective of the gold standard – the relation predicted by a model M for questions Q_i and Q_j does not hold for the answer sets that M itself generates for Q_i and Q_j. In these cases, the model contradicts itself. We describe here some measures and results that we extracted to analyze this issue.

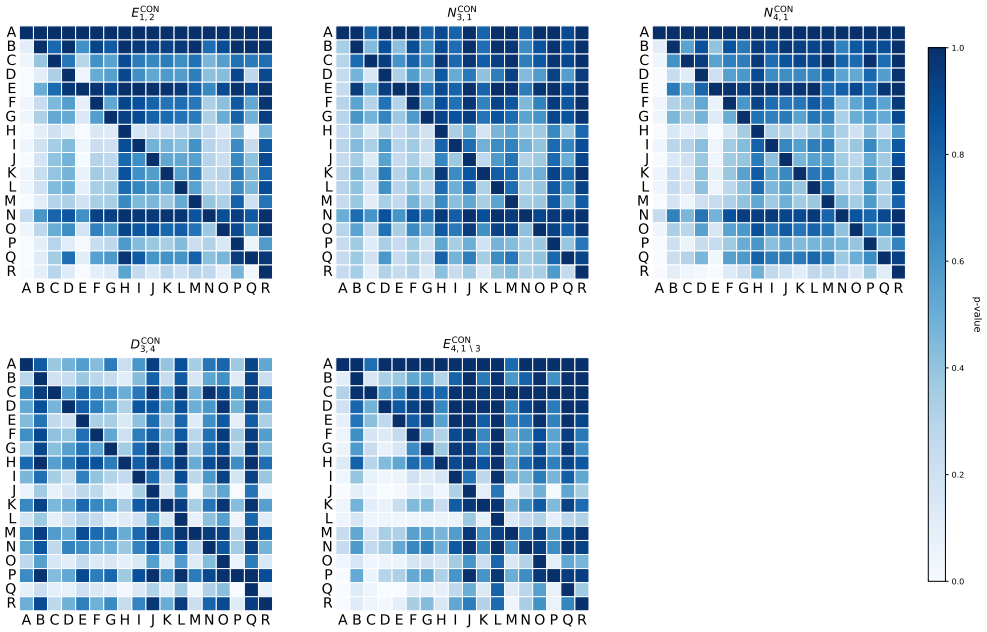

Figure 2: p-value heatmap of LLMs in overall datasets for Base strategy.

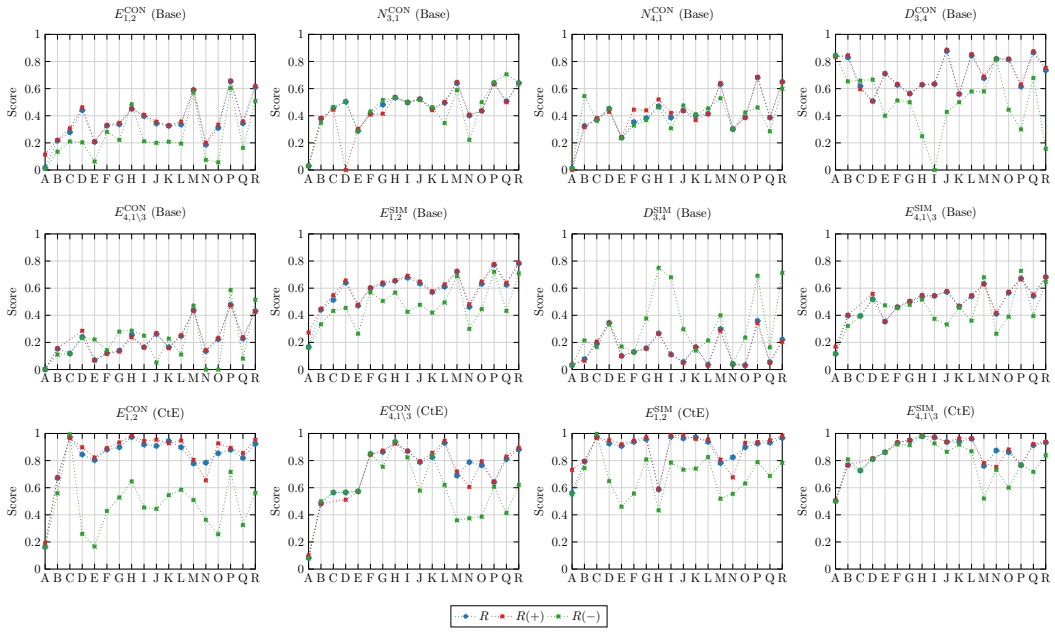

Figure 3: Comparison of R and R(+/–) consistency for Base (1st and 2nd rows) and CtE (3rd row).

Table 7: McNemar p-value which shows the statistically significant improvements over the zero-shot prompting with the Classification-then-Enumerate (CtE) strategy and Oracle on the overall dataset. Asterisks denote statistical significance: *** for $p < 0.001$, ** for $p < 0.01$, and * for $p < 0.05$. The empty cell indicates a p-value ≥ 0.05

	CtE					Oracle				
LLM	$E_{1,2}$	$N_{3,1}$	$N_{4,1}$	$D_{3,4}$	$E_{4,1\setminus 3}$	$E_{1,2}$	$N_{3,1}$	$N_{4,1}$	$D_{3,4}$	$E_{4,1\setminus 3}$
Llama-3.1-8b	***		***		***	***	*	**	**	***
GPT-oss-20b	***		***		***	***	***	***	***	***
GPT-4.1-nano	***	***	***	***		***	***	***		***
Mistral-small:24b	***		***	***	***	***	***	***	***	***
Llama-3.1-70b	***	***	***	***	***	***	***	***	***	***
Gemini-2.0-flash	***	***	***		***	***	***	***	***	***
GPT-4.1-mini	***	*	***		***	***	***	***	***	***
GPT-4o	***	***	***	***	***	***	***	***	***	***
GPT-4.1	***	***	***	***	***	***	***	***	***	***
Grok-3-mini	***	*	***		***	***	***	***	***	***
DeepSeek-V3.1	***	***	***	***	***	***	***	***	***	***
Gemini-2.5-flash	***	***	***		***	***	***	***	***	***
GPT-5-nano	***		***			***	***	***	***	***
DeepSeek-reasoner	***	***	***		***	***	***	***	***	***
Gemini-2.5-pro	***	***	***	***	***	***	***	***	***	***
GPT-5-mini	***			***	***	***	***	***	***	***
GPT-o3	***	***	***	*	***	***	***	***	***	***
GPT-5	***		***		***	***	***	***	***	***

Table 8: Pearson correlations between external model scores and reasoning performance metrics. Asterisks indicate statistical significance: * $p < 0.05$, ** $p < 0.001$.

	$E^{\text{CON}}_{1,2}$	$N^{\text{CON}}_{3,1}$	$N^{\text{CON}}_{4,1}$	$D^{\text{CON}}_{3,4}$	$E^{\text{CON}}_{4,1\setminus 3}$	CON()	$E^{\text{SIM}}_{1,2}$	$D^{\text{SIM}}_{3,4}$	$E^{\text{SIM}}_{4,1\setminus 3}$
r	0.03	0.35	0.29	0.41	0.85**	0.44	0.03	-0.16	0.61**
p	0.91	0.21	0.29	0.14	<0.001	0.10	0.89	0.57	<0.02

K.1 Measures

We consider the following measures to quantify the self-contradictions of the LLMs in this setting.

Contradiction-free rates To complement classification accuracy and answer-set consistency, we measure whether a model is *internally consistent* with respect to the logical relation it *predicts*. Given two questions Q_i, Q_j, We say the case is *contradiction-free* iff the predicted relation is satisfied by the model's *own* answers; otherwise, a *self-contradiction* occurs. We define the *self-contradiction rate* as the percentage of cases where a model's predicted relation is not satisfied by its own answers. This measure distinguishes between models that are internally consistent but wrong and those that are internally inconsistent, thereby offering a finer-grained perspective on model reliability. We denote such rates for a model M and relation R as $R^{\text{CFR}}(M)$.

Consistency by relation correctness To further analyze model behavior, we distinguish consistency depending on whether the logical relation between two questions is correctly identified with respect to the gold standard. This breakdown reveals how much consistency stems from correctly recognizing the relation versus how much persists even when the relation is misclassified.

K.2 Results

In Table 9, we present the results of the contradiction-free rates for five relations. We see that the models do tend to contradict themselves, i.e., the answers they return for questions do not respect

Table 9: Per-relation contradiction-free rates R^{CFR} (0–100 scale).

LLM	Str	$E^{\mathrm{CFR}}_{1,2}$	$N^{\mathrm{CFR}}_{3,1}$	$N^{\mathrm{CFR}}_{4,1}$	$D^{\mathrm{CFR}}_{3,4}$	$E^{\mathrm{CFR}}_{4,1\setminus 3}$
Llama-3.1-8b	Base	93.83	86.33	**87.33**	28.50	97.17
	CtE	81.00	13.83	–"–	–"–	82.67
GPT-oss-20b	Base	26.17	40.33	32.17	80.33	16.00
	CtE	66.50	89.33	–"–	–"–	40.33
GPT-4.1-nano	Base	45.67	50.00	53.17	50.00	96.00
	CtE	69.33	36.50	–"–	–"–	**100.00**
Mistral-small:24b	Base	48.98	49.49	54.58	50.68	87.29
	CtE	88.50	0.17	–"–	–"–	56.50
Llama-3.1-70b	Base	23.00	68.17	62.17	71.17	4.33
	CtE	82.33	9.00	–"–	–"–	53.67
Gemini-2.0-flash	Base	37.33	44.50	62.33	62.50	23.33
	CtE	88.50	46.83	–"–	–"–	46.17
GPT-4.1-mini	Base	38.67	46.00	59.17	56.33	8.67
	CtE	89.33	44.83	–"–	–"–	62.50
GPT-4o	Base	45.33	49.00	53.50	63.17	37.67
	CtE	**96.50**	66.33	–"–	–"–	73.50
GPT-4.1	Base	42.83	49.83	50.33	63.83	10.83
	CtE	92.33	69.67	–"–	–"–	74.67
Grok-3-mini	Base	39.33	52.17	44.00	**88.00**	26.17
	CtE	91.83	95.17	–"–	–"–	69.33
DeepSeek-V3.1	Base	34.83	50.33	54.33	56.00	14.83
	CtE	89.33	10.00	–"–	–"–	52.00
Gemini-2.5-flash	Base	41.83	50.83	41.83	84.00	25.67
	CtE	87.50	94.83	–"–	–"–	85.33
GPT-5-nano	Base	57.83	61.33	62.67	65.83	24.50
	CtE	77.67	89.33	–"–	–"–	8.83
DeepSeek-reasoner	Base	28.17	41.83	31.50	79.67	16.50
	CtE	65.33	92.00	–"–	–"–	54.00
Gemini-2.5-pro	Base	38.67	43.67	39.67	81.67	23.33
	CtE	90.67	95.83	–"–	–"–	74.67
GPT-5-mini	Base	63.33	63.00	68.33	63.50	19.17
	CtE	84.67	96.17	–"–	–"–	38.00
GPT-o3	Base	39.32	49.49	39.66	84.92	25.42
	CtE	84.33	85.67	–"–	–"–	77.50
GPT-5	Base	60.83	63.50	64.67	75.67	26.83
	CtE	91.33	**98.17**	–"–	–"–	63.00

the relation between the questions that they themselves predict. A lot of variance is seen across the models, with large models showing more consistency.

Furthermore, we examine how consistency varies depending on whether the relation classification is correct. Table 10 presents the results for consistency and similarity when the predicted relation is correct, while Table 11 shows the corresponding results when the relation is incorrect. Figure 3 compares consistency across all cases, as well as positive and negative subsets. The results indicate that incorrect relation classification generally leads to lower consistency.

Table 10: Per-relation consistency rate and similarity when predicted relation is correct according to ground-truth (0–100 scale)

LLM	Act	$E^{\text{CON}}_{1,2}$	$N^{\text{CON}}_{3,1}$	$N^{\text{CON}}_{4,1}$	$D^{\text{CON}}_{3,4}$	$E^{\text{CON}}_{4,1\setminus 3}$	$E^{\text{SIM}}_{1,2}$	$D^{\text{SIM}}_{3,4}$	$E^{\text{SIM}}_{4,1\setminus 3}$
Llama-3.1-8b	Base	11.43	2.94	0.00	83.48	0.00	27.31	3.64	16.82
	CtE	19.23	**100.00**	–"–	–"–	10.26	73.28	–"–	50.77
GPT-oss-20b	Base	22.20	38.27	31.66	84.67	15.60	44.83	6.45	40.29
	CtE	67.84	**100.00**	–"–	–"–	48.26	79.69	–"–	76.58
GPT-4.1-nano	Base	30.84	44.49	38.31	59.63	–"–	54.88	20.64	–"–
	CtE	96.51	66.23	–"–	–"–	–"–	96.72	–"–	–"–
Mistral-small:24b	Base	46.21	0.00	42.86	50.77	28.57	65.68	**34.47**	55.83
	CtE	89.82	**100.00**	–"–	–"–	51.16	95.08	–"–	81.67
Llama-3.1-70b	Base	21.06	30.43	24.07	71.26	6.77	47.69	10.00	35.24
	CtE	82.30	**100.00**	–"–	–"–	57.27	92.09	–"–	86.26
Gemini-2.0-flash	Base	33.21	41.03	44.53	63.46	11.60	60.45	13.01	46.20
	CtE	89.25	**100.00**	–"–	–"–	84.55	94.75	–"–	93.57
GPT-4.1-mini	Base	34.80	41.55	44.00	56.40	13.39	64.19	15.52	50.27
	CtE	93.42	65.32	–"–	–"–	87.39	97.17	–"–	95.15
GPT-4o	Base	44.97	53.68	52.05	63.09	23.79	65.84	26.36	55.03
	CtE	**98.28**	99.25	–"–	–"–	92.57	59.12	–"–	**98.35**
GPT-4.1	Base	40.74	49.83	42.11	63.65	16.44	69.17	10.90	54.46
	CtE	94.53	70.34	–"–	–"–	87.14	98.65	–"–	97.24
Grok-3-mini	Base	35.64	52.37	43.70	**88.74**	26.85	64.90	5.04	57.93
	CtE	95.42	**100.00**	–"–	–"–	79.52	**98.67**	–"–	93.89
DeepSeek-V3.1	Base	32.99	44.13	36.80	56.04	15.84	57.66	16.63	46.73
	CtE	92.81	**100.00**	–"–	–"–	85.71	95.98	–"–	96.43
Gemini-2.5-flash	Base	35.71	50.17	41.35	85.37	25.31	62.83	3.01	54.82
	CtE	94.79	**100.00**	–"–	–"–	**94.57**	95.60	–"–	96.53
GPT-5-nano	Base	59.17	64.90	64.12	68.93	43.33	72.58	28.56	62.84
	CtE	80.55	**100.00**	–"–	–"–	72.00	80.81	–"–	78.26
DeepSeek-reasoner	Base	20.08	40.72	30.17	82.01	14.31	48.13	3.94	41.90
	CtE	65.47	**100.00**	–"–	–"–	60.56	67.65	–"–	75.33
Gemini-2.5-pro	Base	33.39	43.52	38.62	82.06	23.22	64.90	2.95	57.36
	CtE	92.70	**100.00**	–"–	–"–	79.50	93.07	–"–	88.22
GPT-5-mini	Base	65.87	63.81	**68.65**	63.16	47.29	77.73	34.18	66.80
	CtE	89.35	100.00	–"–	–"–	64.34	93.71	–"–	76.66
GPT-o3	Base	35.83	50.09	38.89	87.54	23.87	63.97	4.98	55.25
	CtE	85.54	88.41	–"–	–"–	83.01	94.99	–"–	92.23
GPT-5	Base	62.06	64.16	65.08	75.39	42.68	78.91	20.50	68.34
	CtE	95.47	**100.00**	–"–	–"–	89.49	98.21	–"–	93.82

L Causal Analysis of LLMs Inconsistency

As we observed, LLMs often give varying responses to identical inputs, potentially due to four factors: decoding randomness, computational nondeterminism, order sensitivity, and data-level conflicts.

L.1 Decoding Randomness

The most immediate cause of nondeterminism in LLM output is the probabilistic nature of token generation. The sampling temperature T (Ackley et al., 1985; Li et al., 2025) acts as a hyperparameter that influences the output distribution. Formally, the probability of selecting token w_i is: $P(w_i) =$

Table 11: Per-relation consistency rate and similarity when predicted relation is incorrect according to ground-truth (0–100 scale)

LLM	Act	$E^{\text{CON}}_{1,2}$	$N^{\text{CON}}_{3,1}$	$N^{\text{CON}}_{4,1}$	$D^{\text{CON}}_{3,4}$	$E^{\text{CON}}_{4,1\setminus 3}$	$E^{\text{SIM}}_{1,2}$	$D^{\text{SIM}}_{3,4}$	$E^{\text{SIM}}_{4,1\setminus 3}$
Llama-3.1-8b	Base	1.06	3.01	1.32	**84.54**	0.17	16.01	3.21	11.58
	CtE	16.20	**100.00**	–"–	–"–	8.05	55.25	–"–	50.09
GPT-oss-20b	Base	13.51	34.78	54.55	65.38	11.11	33.38	21.46	32.12
	CtE	55.88	**100.00**	–"–	–"–	50.00	74.50	–"–	80.90
GPT-4.1-nano	Base	21.08	46.43	36.36	65.93	11.83	43.23	16.67	39.49
	CtE	**99.41**	73.50	–"–	–"–	56.50	**99.71**	–"–	72.72
Mistral-small:24b	Base	20.41	50.42	45.28	66.67	23.46	45.42	33.33	51.40
	CtE	26.00	**100.00**	–"–	–"–	56.91	64.90	–"–	81.06
Llama-3.1-70b	Base	6.25	28.70	23.74	40.00	22.22	26.46	17.00	47.39
	CtE	16.67	**100.00**	–"–	–"–	57.52	46.07	–"–	86.09
Gemini-2.0-flash	Base	28.12	43.18	32.84	51.28	14.29	56.97	12.86	45.45
	CtE	42.86	**100.00**	–"–	–"–	85.25	55.66	–"–	91.98
GPT-4.1-mini	Base	22.22	51.65	36.84	50.00	28.00	50.59	37.72	47.75
	CtE	52.83	63.47	–"–	–"–	75.47	80.86	–"–	91.40
GPT-4o	Base	48.48	53.17	46.30	25.00	28.71	56.77	**75.00**	51.64
	CtE	64.71	99.50	–"–	–"–	**94.12**	43.37	–"–	**97.87**
GPT-4.1	Base	21.21	50.00	30.77	0.00	25.00	42.59	68.06	37.38
	CtE	45.45	70.00	–"–	–"–	82.35	78.50	–"–	92.73
Grok-3-mini	Base	20.00	51.61	47.62	42.86	5.26	47.73	29.75	33.28
	CtE	44.44	**100.00**	–"–	–"–	57.89	73.36	–"–	86.46
DeepSeek-V3.1	Base	20.83	46.25	41.05	50.00	22.81	41.98	14.06	45.65
	CtE	54.55	**100.00**	–"–	–"–	83.92	74.09	–"–	91.83
Gemini-2.5-flash	Base	19.51	34.62	45.45	57.89	11.11	49.53	21.45	36.10
	CtE	58.54	**100.00**	–"–	–"–	62.07	82.66	–"–	86.86
GPT-5-nano	Base	57.14	58.89	52.94	57.97	47.17	68.94	40.02	68.06
	CtE	50.91	**100.00**	–"–	–"–	36.00	51.92	–"–	52.07
DeepSeek-reasoner	Base	7.46	22.22	30.00	81.82	0.00	29.98	4.50	26.44
	CtE	36.36	**100.00**	–"–	–"–	37.50	55.45	–"–	73.20
Gemini-2.5-pro	Base	5.77	50.00	42.42	44.44	0.00	44.58	23.57	39.00
	CtE	25.76	**100.00**	–"–	–"–	38.64	63.17	–"–	60.09
GPT-5-mini	Base	60.34	64.71	46.15	30.00	58.62	72.10	69.17	72.76
	CtE	71.74	**100.00**	–"–	–"–	60.71	78.90	–"–	77.00
GPT-o3	Base	16.28	70.59	28.57	67.86	8.11	43.24	16.46	39.48
	CtE	32.50	86.36	–"–	–"–	41.38	68.66	–"–	71.69
GPT-5	Base	50.88	64.29	**60.00**	15.79	51.52	71.02	71.31	64.70
	CtE	56.25	**100.00**	–"–	–"–	62.07	78.48	–"–	84.04

$\frac{e^{z_i/T}}{\sum_j e^{z_j/T}}$, where z_i denotes the logit for token w_i. When $T > 0$, sampling methods such as top-k and nucleus (top-p) sampling apply stochasticity to improve diversity, ensuring that identical inputs can yield semantically distinct outputs. Theoretically, setting $T = 0$ (greedy decoding) should guarantee determinism. However, empirical studies (Li et al., 2025; Renze, 2024; Wang & Wang, 2025; Atıl et al., 2025) demonstrate that outputs often remain unstable even with the temperature set to zero, pointing to other underlying causes of nondeterminism.

L.2 Computational Nondeterminism

Nondeterminism at $T = 0$ can arise from the non-associative nature of floating-point arithmetic in GPUs, where $(a + b) + c \neq a + (b + c)$ due to rounding errors. As Yuan et al. (2025) demonstrates,

minor numerical discrepancies in early tokens can cascade into divergent outputs, particularly in reasoning-heavy tasks. Furthermore, modern optimizations, such as Mixture-of-Experts (MoE) (Masoudnia & Ebrahimpour, 2014), FlashAttention (Dao et al., 2022), and continuous batching often process data in non-deterministic orders to maximize throughput. Atıl et al. (2025) highlight that while fixing random seeds can mitigate multi-GPU randomness, engineering optimizations (e.g., chunk prefilling or prefix caching) frequently reintroduce nondeterministic behavior, making exact reproducibility fragile even in controlled environments.

L.3 Order Sensitivity

LLMs exhibit hypersensitivity to input presentation due to the self-attention mechanism (Vaswani et al., 2017). Because attention weights rely on positional encodings, trivial formatting changes – such as reordering few-shot examples or altering whitespace – can have notable effects on output. Liu et al. (2024a) identifies a "lost in the middle" phenomenon, where models fail to access information located in the middle of a long context window. Similarly, Lu et al. (2022) shows that the chosen order of few-shot examples can change performance from near-random to state-of-the-art.

L.4 Data-Level Conflicts

Nondeterminism may also arise due to differences between pre-trained weights (parametric memory) and the provided context (non-parametric memory). Xu et al. (2024) identifies knowledge conflicts, where externally retrieved context contradicts internal training data, as a source of nondeterminism. Another issue is due to temporal misalignment; Nakshatri et al. (2025) finds that while models may contain updated knowledge, they often struggle to retrieve it reliably against outdated but strongly-weighted internal facts. Consequently, models may switch in an unpredictable manner between "local optima," prioritizing helpfulness over correctness (Xie et al., 2024).

M Inconsistency Error Analysis

We conducted a detailed analysis of the errors that continue to cause inconsistencies in the LLMs responses, even when applying the **CtE** and **Oracle** prompting strategies. We identified the following recurring error patterns, which can be classified into the categories listed below, excluding those errors attributable to the stochastic nature of the model, which have already been discussed in Appendix L.

- Use of different terminology to refer to the same concept or entity: In some questions involving country lists, the model refers to the same country using different formulations, for instance, "Spain" versus "Kingdom of Spain". This occurs despite our prompting guidelines instructing the LLM to provide full names and avoid abbreviations. The number of answers affected by this problem is reported in Appendix D.
- Completeness of the response: For the case of relation $E_{1,2}$, we observed instances in which the same question produced answers with different set cardinalities across different executions.
- Incapability of capturing implicit logic: As discussed in Appendix K.2, LLMs that misclassify the relation often show lower consistency in subsequent answers. When the relation is incorrect, follow-up reasoning becomes unreliable. This highlights that the ability to correctly grasp the implicit logic and contextual relationships behind the questions is crucial for achieving higher consistency.
- Lack of option for uncertainty: The strategy of permitting LLMs to answer '`idk`' when uncertain generally improves consistency. Without this option, LLMs may attempt to answer despite uncertainty, often leading to hallucinated responses that cause inconsistency.

N GenAI Usage Disclosure

GenAI was used for text refinement, to assist code debugging, and for dataset creation (as described in the submission).