# Exploring Relational Reasoning Capabilities in LLMs with REL

**Lukas Fesser** [*1] **Yasha Ektefaie** [*1] **Ada Fang** [*1] **Marinka Zitnik** [1]

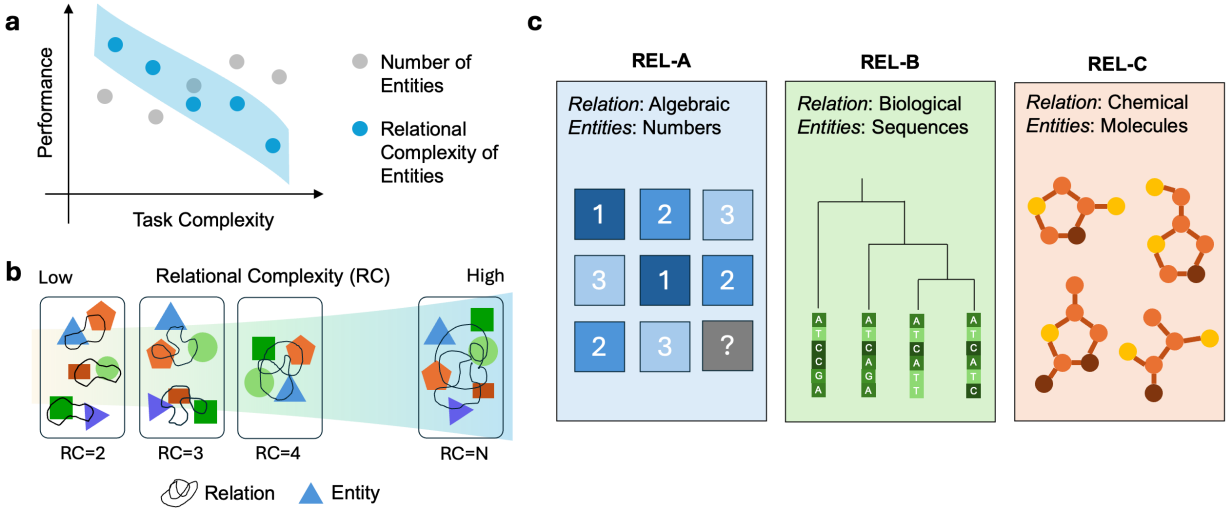


*Figure 1:* **a** When tasks are stratified by relational complexity, performance degrades with relational complexity even when the number of entities in the task varies. Entity count is a noisy proxy for task difficulty. **b** Relational complexity increases with the size of the set that must satisfy a relation collectively under shared constraints, i.e., when correctness depends on a higher-arity constraint. **c REL** explores capabilities of LLMs to perform relational reasoning across algebraic, biological and chemical domains.

## Abstract

Relational reasoning is the ability to infer relations that jointly bind multiple entities, attributes, or variables. While this capability is essential for scientific reasoning, most existing evaluations of relational reasoning in large language models focus on structured inputs such as tables, graphs, or synthetic relational tasks, and do not isolate the sources of difficulty that arise from higher-arity relational binding. We study this problem through the lens of *Relational Complexity (RC)*, defined as the minimum number of independent entities or operands that must be simultaneously bound to apply a relation. RC provides a principled way to vary reasoning difficulty independently of confounders such as input size, vocabulary, and representational choices. Building on RC, we introduce **REL**, a generative benchmark framework spanning algebra, chemistry, and biology that varies RC within each domain. Evaluating frontier LLMs, we observe a consistent and monotonic degradation in performance as RC increases, even when the total number of entities is held fixed. This failure mode persists under increased test-time compute and with in-context learning, suggesting a limitation tied to the arity of the required relational binding rather than insufficient inference steps or exposure to examples. Our results identify a well-defined regime of higher-arity reasoning in which current models struggle, and motivate revisiting reasoning benchmarks through the lens of relational complexity.

GitHub Hugging Face

## 1. Introduction

Relational reasoning, inferring an unknown from the structured interaction of multiple entities and relations, is widely assumed to be a core component of human reasoning capabilities (Halford et al., 1998; Alexander et al., 2016; Dumas et al., 2013). Yet despite the considerable recent progress of Large Language and Reasoning Models (LLMs and LRMs) on a variety of reasoning tasks, relational reasoning is rarely assessed directly (Clark et al., 2018; Cobbe et al., 2021; Hendrycks et al., 2021; Geva et al., 2021; Liévin et al., 2024; Wang et al., 2023b). Research that explicitly targets relational reasoning in these models tends to concentrate on graph-based settings over explicit graphs or knowledge

[1]Harvard University. Correspondence to: Marinka Zitnik <marinka@hms.harvard.edu>.

*Preprint. February 9, 2026.*

graphs (Wang et al., 2023a; Tang et al., 2024; Wu et al., 2025; Zhang et al., 2024; Fatemi et al., 2024), or multi-hop reasoning over knowledge bases (Yang et al., 2018; Ho et al., 2020; Trivedi et al., 2022). This leaves open a broader question: *how well do state-of-the-art LLMs/LRMs handle relational structure in scientific reasoning*, such as numerical pattern rules, algebraic constraints, or chemical structure regularities? We address this question with a unified set of tasks that varies relational complexity while controlling for other sources of difficulty, including entity complexity, across multiple scientific domains.

In many scientific problems, the answer is not carried by any single cue. Instead, a model must combine multiple constraints that link variables, measurements, or symbols to reach a correct conclusion (Bemis & Murcko, 1996; Stern, 2013). When we cannot measure where models break under this multi-constraint integration, benchmark performance becomes difficult to interpret: strong results may reflect saturation rather than progress on harder forms of reasoning (Deveci & Ataman, 2025). Identifying unsaturated dimensions of reasoning performance is necessary to understand where current models still fail and where further improvements are possible. Evaluating this capability is challenging because standard proxies for "difficulty" do not isolate the relational bottleneck. Performance can drop for reasons that have little to do with relational reasoning, such as longer prompts, different prompt templates, or additional background knowledge being required. As a result, evaluations often mix the cost of parsing and retrieval with the cost of relational inference. Existing studies have largely evaluated relational reasoning through graph-centric formulations (Liu et al., 2025a). Beyond graph-centric tasks, we still lack a task-agnostic notion of relational difficulty.

Here we introduce *Relational Complexity* (RC) as a principled measure of how many independent operands and entities must be represented and jointly composed to solve a task (Fig. 1b). Importantly, relational complexity has a history outside of machine learning, where it is used in cognitive science and related fields (Halford et al., 1998; Carpenter et al., 1990; Crone et al., 2009) to characterize reasoning demands. In this paper, we present **REL**, a suite of tasks where relational complexity can be controlled. Our approach has the following advantages: **(1) Explicit and parameterization of RC for scientific reasoning.** We design tasks across mathematics, biology, and chemistry that allow systematic control over relational complexity (Fig. 1c). **(2) Isolating the effect of RC on performance.** We isolate the impact of RC on LLM performance by controlling for and marginalizing over other confounding factors. **(3) A scalable framework for probing scientific relational reasoning.** We introduce a generative benchmark that produces novel questions at systematically increasing levels of difficulty, enabling evaluation across model capabilities.

Evaluating frontier LLMs on **REL**, we observe that performance degrades sharply as relational complexity increases across mathematical, biological, and chemical domains. Measures of RC are complementary to performance changes observed in size-based proxies including input length and entity count (Fig. 1a). For Raven's matrices and tensors in **REL**-A, performance drops by 45% when RC increases from 3 to 9. Applying RC to reasoning over evolutionary history with **REL**-B, we find increasing RC from 5 to 20 leads to accuracy decreasing by 93%. In chemical relational reasoning task completion rate for reasoning over SMILES drops by 39.7% from **REL**-C1 to **REL**-C3. Relational complexity captures a distinct and consequential barrier for current state-of-the-art models.

## 2. Related Work

**Reasoning Benchmarks in Machine Learning.** Reasoning benchmarks in machine learning have expanded rapidly alongside the development of LLMs and LRMs. Existing benchmarks span a wide range of capabilities, including arithmetic and symbolic manipulation (Hendrycks et al., 2021; Cobbe et al., 2021), program synthesis (Austin et al., 2021; Chen, 2021), logical deduction (Clark et al., 2018; Liu et al., 2020; Tafjord et al., 2021), tool use (Qin et al., 2024; Li et al., 2023), and multi-hop question answering (Yang et al., 2018; Ho et al., 2020; Trivedi et al., 2022). A key limitation of existing benchmarks is the lack of explicit control over the relational structure underlying a task. Recent work has begun to probe LLMs' ability to reason over graphs (Wang et al., 2023a; Tang et al., 2024; Wu et al., 2025; Zhang et al., 2024; Fatemi et al., 2024) and knowledge graphs (Edge et al., 2024; Zhu et al., 2025; He et al., 2024; Li et al., 2024; Luo et al., 2023). Liu et al. (2025a) introduces a generative benchmark for relational reasoning, but their framework focuses on changing the underlying relational graph structure of tasks. While valuable, these settings rely on graph-specific formalisms that do not generalize cleanly to broader scientific reasoning tasks and often vary surface representations without systematically altering the underlying relational complexity.

**Relational Reasoning in Cognitive Science.** In cognitive science, relational reasoning is commonly studied as the ability to represent relations among entities, align roles across multiple structures, and compose several relations to infer a missing element or choose a consistent completion (Alexander et al., 2016; Dumas et al., 2014; Carpenter et al., 1990; Halford et al., 1998; Crone et al., 2009). Tasks such as Raven's Progressive Matrices and related analogical paradigms are widely used (Brouwers et al., 2009; Mills et al., 1993; Burke, 1972; Williams & McCord, 2006) because they require extracting abstract relational structure that generalizes beyond surface features, and they permit

controlled manipulations of the number of relations that must be simultaneously maintained. A closely related concept is relational complexity (Halford et al., 1998), which characterizes task demand by the number of independent "slots" (entities or variables) that must be bound and processed concurrently to perform the required inference. This framing separates relational difficulty from superficial complexity and helps explain sharp changes in performance as tasks move from binary to higher-arity relational integration. In this paper, we leverage it to motivate a model-agnostic difficulty axis for evaluating LLM/LRM relational reasoning. We provide an extended related work in Appendix B.

## 3. Defining Relational Complexity

Here we formalize the concept of Relational Complexity using the example of Raven's Progressive Matrices (RPMs) (John & Raven, 2003), before introducing the individual components of **REL** which spans arithmetic, biology, and chemistry (Fig. 2).

**Definition of Relational Complexity (RC).** Relational complexity (RC) is the minimal number of independent sources of variation that must be bound and represented at the same time to carry out a reasoning step (Fig. 1b). Independent sources are variables that can change freely and must be tracked separately. Binding links specific fillers to distinct argument roles, which are the slots a relation provides. The required representations span as many dimensions as there are such sources, and the relational complexity equals the relation's arity, meaning the number of argument roles that must be handled together.

**Definition of Operand Complexity (OC).** Operand complexity (OC) refers to the difficulty of identifying, representing, or inferring the fillers that occupy argument roles in a relation, independent of the number of roles themselves. Tasks can share the same RC but have different OC.

We use **Raven's Progressive Matrices** (RPMs) (John & Raven, 2003) to illustrate this definition. In their original form, RPMs are tasks that associate vision with relational and analogical reasoning in a hierarchical representation. They come with three different levels of difficulty defined by RC, and were introduced to test cognitive abilities in children and young adults (Carpenter et al., 1990). Consider the three examples of RPMs in Fig. 3, below we describe how we obtain the RC of each matrix.

In $\text{RPM}_1$ of Fig. 3 no row or column relation is present, only matching. The solver holds a single template feature (e.g., "solid upright triangle") and scans the three candidates until one is identical. Only one independent source of variation, the template itself, must be represented in parallel to determine the answer. Hence, the bottleneck is unary, giving $\text{RC} = 1$.

In $\text{RPM}_2$ of Fig. 3 the rule is "horizontal or vertical," so the blank can be solved by using one axis only, either the row or the column. For any active attribute (here, the semicircle's orientation), the two known cells along the chosen axis determine the third. Thus, the bottleneck is binary, giving $\text{RC} = 2$.

In $\text{RPM}_3$ of Fig. 3 the missing cell must satisfy both the row rule and the column rule at the same time. For any active attribute (e.g., the symbol's type or marking), the value in the blank is determined by integrating the two known cells in its row with the two known cells in its column. Four independent operands must be held in parallel, so the bottleneck is quaternary, giving $\text{RC} = 4$.

## 4. Relational Reasoning Benchmark

### 4.1. Relational Reasoning in Algebra (REL-A)

RPMs do not need to involve visual components, and we can reduce them to symbolic tasks by representing matrices using their attributes, e.g. "a black triangle in column one, row one" as done in (Hersche et al., 2025), or by working directly with numerical arrays as in Fig. 2 (Camposampiero et al., 2025b). Unlike (Camposampiero et al., 2025a;b), we do not use confounders or noise to confuse the model since we are primarily interested in relational reasoning capabilities as measured via relational complexity, which neither confounders nor perceptual noise affect. We note that the missing entry in an RPM is a function of the remaining entries, so we can directly control the relational complexity of the task by increasing the number of entries on which that function's output depends.

To design more difficult RPMs with a relational complexity much greater than 4, we introduce a generalization we call **Raven's Progressive Tensors (RPT)s**. To solve the RPT, models must reason over functions and rules where the missing value depends on the one-hop neighborhood, such as the sum of adjacent values. With higher dimensions, we can achieve $\text{RC}_{2-\text{dim}} \leq 8$, $\text{RC}_{3-\text{dim}} \leq 26$, $\text{RC}_{4-\text{dim}} \leq 80$, $\ldots$, $\text{RC}_{n-\text{dim}} \leq 3^n - 1$. In our experiments, we use the following seven rules to generate RPMs and RPTs with varying relational complexities.

Suppose we are given a $n \times n$ RPM: **A1 (Constant).** Each entry in the RPM are the same value. $\text{RC} = 1$. **A2 (Progression).** Each entry is the value of its predecessor in the same row plus a fixed value. $\text{RC} = 2$. **A3 (Permutation).** Each row contains the same $n$ values in a random (non-repeating) order. $\text{RC} = n$. **A4 (Row-Sum).** The final value in each row is the sum of all other entries in the same row multiplied by either $\pm 1$, depending on the column. $\text{RC} = n$. Now suppose we are given a $n \times n \times n$ RPT: **A5 (4-Moving-Average).** Each entry is the sum of the same 4 predecessors

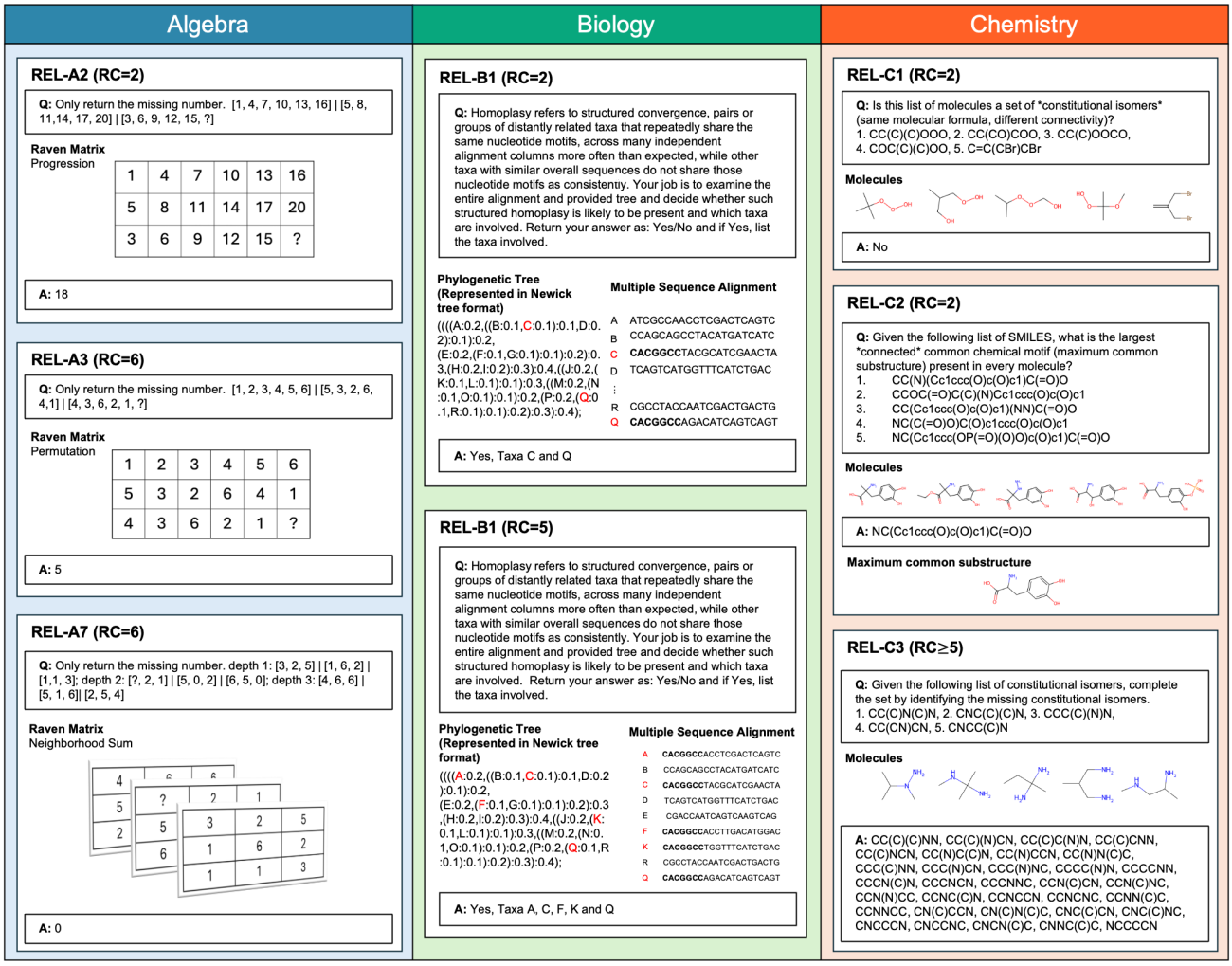


*Figure 2.* **REL** evaluates relational reasoning across algebraic, biological, and chemical relations.

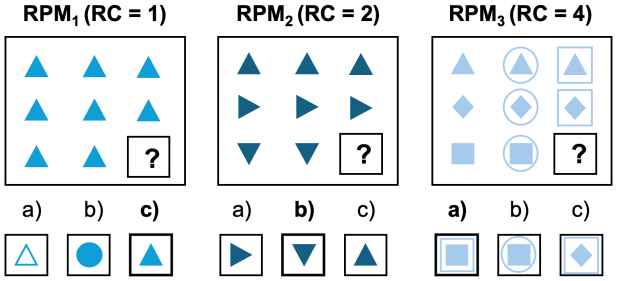


*Figure 3.* Three examples of Raven's Progressive Matrices with increasing relational complexity. The answers are shown in bold.

along the $x$-, $y$-, and $z$-axis. RC $= 4$. **A6 (5-Moving Average).** Same as previous, but with 5 predecessors. RC $= 5$. **A7 (Neighborhood Sum).** Each entry of the RPT is the sum of its neighbors modulo 7. RC $= 6$ in a $3 \times 3 \times 3$ tensor.

We use the same set up as in the original vision-based RPM task, the model needs to determine the missing value based on the given matrix or tensor. We detail the generation of **REL**-A in Appendix A.1. The resulting **REL**-A dataset consists of 3,500 RPMs/ RPTs in total, but the synthetic dataset generators introduced here allow for the construction of potentially many more **REL**-A type questions with various sizes and value ranges.

### 4.2. Relational Reasoning in Biology (REL-B)

There exists many benchmarks for biological sequences, including ProteinGym (Notin et al., 2023) that evaluates whether a model can predict the effect of individual variants in protein sequences, DNALongBench (Cheng et al., 2025) that evaluates whether a model can predict the long-range effects of genomic elements, and TAPE (Rao et al., 2019)/PEER (Xu et al., 2022) with diverse tasks such as protein-protein interaction and protein localization prediction. These benchmarks typically evaluate tasks defined over individual sequences or sequence pairs (Rong et al., 2025; Gao et al., 2025; Ye et al., 2024).

However, many biological inferences fundamentally require relational comparisons across multiple organisms condi-

tioned on their evolutionary history, such as detecting convergent evolution or homoplasy (Wake et al., 2011), where similar motifs arise independently in distinct lineages (Appendix B.2). To evaluate a model's ability to detect homoplasy we provide a model with an multiple sequence alignment (MSA) and the corresponding phylogenetic tree and ask it to (1) decide whether homoplasy is present and (2) identify the taxa participating in the homoplastic motif (Fig. 4). Solving the task requires jointly (a) localizing a shared motif across sequences and (b) verifying from the tree that the motif spans evolutionarily distinct lineages rather than a single recent clade. Here RC $= N_{ht}$, where $N_{ht}$ is the number of homoplastic taxa, as the model must maintain in its memory the current position of the taxa in relation to the other homoplastic taxa in the tree.

To systematically evaluate this capability, we construct a synthetic dataset generator parameterized by four variables: (1) the number of homoplastic taxa $N_{ht}$, (2) the number of leaves in the phylogenetic tree $N_{\text{leaves}}$, (3) the sequence length $L_{\text{seq}}$, and (4) the length of the conserved motif $L_{\text{motif}}$ (Figure 4). This construction enables the generation of a large and diverse set of questions spanning a wide range of difficulty regimes, while preserving a known ground truth for both homoplasy presence and taxon identity. In particular, the generator allows us to scale the dataset combinatorially across parameter settings, producing many distinct MSAs and trees without manual curation. Notably, **REL**-B is scalable to various levels of RC as we can adjust the number of homoplastic taxa. For the evaluations below we generated 2,600 questions. Further details on question generation and parameters used are available in Appendix A.2.

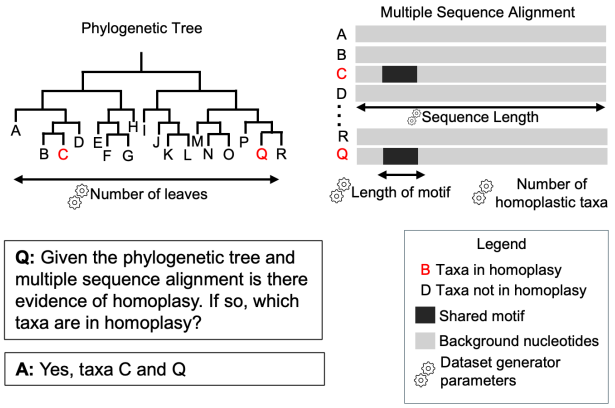


*Figure 4.* From provided parameters, we generate a phylogenetic tree, alignment, inject shared motifs, and ask the model to use this alignment and tree to return the taxa that are in homoplasy.

### 4.3. Relational Reasoning in Chemistry (REL-C)

Relational reasoning between molecules is key to understanding molecular function for systematic exploration of vast chemical spaces (Bemis & Murcko, 1996). This skill is central to library design (Hajduk et al., 2011) and the analysis of structure-activity relationships (Guha, 2013) in drug discovery. Chemically meaningful inferences rely on shared functional motifs between molecules. For example, molecules which share an aromatic ring, such as a phenyl group, often exhibit similar stacking interactions and hydrophobic character. In contrast, current approaches focus on evaluating molecular comprehension and editing of one molecule or combining molecules in a reaction (Fang et al., 2023; Jang et al., 2025; Runcie et al., 2025; Li et al., 2025; Liu et al., 2025b). Here we introduce three tasks where the entities consist of molecules and tasks involve resolving relations of increasing complexity between the molecules.

**C1: Constitutional isomer set classification.** This question evaluates binary classification of whether a set of molecules forms a constitutional isomer set, molecules that have the same molecular formula but different bond connectivity. Here RC $= 2$ as the model maintains (1) the current shared molecular formula and (2) the formula of the current molecule in the set to iteratively confirm if all molecules share the same molecular formula. For this task, we construct isomers by sampling molecular formulae spanning $C_{3-9}$ with heteroatoms (O, N, S, F, Cl, Br) and various degrees of unsaturation. For "No" instances, we sample $n-1$ molecules from one formula and 1 molecule from a different formula. In total 1,000 questions were curated for this task, with 100 questions for each $N_{\text{molecules}} \in \{5, 10, 15, 20, 25, 30, 35, 40, 45, 50\}$ (Appendix A.3.1). For **REL**-C1, the task completion rate is defined as the accuracy of the responses.

**C2: Largest continuous common chemical motif.** C2 evaluates the ability to identify the maximum common substructure (MCS) shared across a set of molecules. We construct instances by sampling similar molecules from drug-like compounds sampled from ChEMBL (Mendez et al., 2019). The ground-truth MCS is required to contain at least 8 atoms, is connected, and maximizes the number of bonds. The bottleneck is binary, as the model needs to maintain the current largest common substructure and update it with each new molecule in the set. In total, we generate 1,016 questions (Appendix A.3.2) with $N_{\text{molecules}} \in \{5, 10, 15, 20, 25, 30, 35, 40, 45, 50\}$, the number of questions for each number of molecules is provided in Table A.2.

While both C1 and C2 have binary relational complexity, the operand complexity between a pair of molecules in C2 is more challenging than in C1. In C1, two molecules are related if they share the same chemical formulae which is obtained by counting the number of atoms of each element. Conversely, the relation between two molecules in C2 is determined by the largest common substructure.

Predictions are evaluated using a bidirectional substructure metric. We first compute the fraction of examples where the

prediction is a substructure of the ground truth ($S_{\text{pred}\subseteq\text{true}}$) and the fraction where the ground truth is a substructure of the prediction ($S_{\text{true}\subseteq\text{pred}}$). The final metric is:

$$\text{IsSubstructure} = \frac{1}{2}\left(S_{\text{pred}\subseteq\text{true}} + S_{\text{true}\subseteq\text{pred}}\right) \quad (1)$$

This captures both precision (avoiding extraneous atoms) and completeness (including all correct atoms), providing a more nuanced evaluation than binary accuracy for the maximum common substructure task.

**C3: Missing isomer completion.** C3 evaluates the ability to complete a constitutional isomer family given a partial set of observed molecules. To answer this question, the model must infer the full space of valid constitutional isomers implied by the shared molecular formula and identify which of these structures are not present in the observed set. Unlike C1 and C2, this task cannot be reduced to a serial binary update: determining whether a candidate isomer is missing requires simultaneously binding (1) the shared molecular formula, (2) the molecules in the isomer family, (3) the subset of isomers already observed. The relational complexity of C3 arises from the need to simultaneously bind multiple independently varying sources, including the full isomer space of size $N_{\text{isomers}}$ and an observed subset of size $N_{\text{observed}}$. We generate a total of 1,000 questions, where the average number of isomers to be identified is 29 (Appendix A.3.3).

Across **REL**-C1, C2, and C3, we generate a total of 3,016 questions. Because **REL** is a generative framework, additional questions at fixed levels of relational complexity (RC) can be readily sampled from the molecule bank.

## 5. Experiments

We benchmark Claude Opus 4.5, Gemini 3 Pro Preview, and GPT 5.2 on questions from **REL**.

### 5.1. Algebra Tasks

**Model evaluation.** We provide RPMs to the model using the format "$[a_{1,1}, ..., a_{1,n}] \mid ... \mid [a_{n,1}, ..., a_{n,n}]$", where $[a_{i,1}, ..., a_{i,n}]$ is the $i$-th row of an $n \times n$ input. The location of the missing value that the mode is asked to provide is marked with "?", as in Fig. 2. In accordance with the original cognitive tasks given to adolescents (Carpenter et al., 1990) and with other works using RPMs to test machine learning models (Camposampiero et al., 2025a; Hersche et al., 2025), we give the model multiple (8) different possible answer values to choose from, only one of which is correct, so trivial accuracy is $12.5\%$.

**Results.** We present our evaluation results in Fig. 5. All three models solve the tasks with low RC (**REL**-A1, where RC=1 and **REL**-A2, where RC=2) almost perfectly, with accuracy reaching $91\%$ even on the largest $30 \times 30$ RPMs. On tasks where RC scales with the size of the input (**REL**-A3 and **REL**-A4), all three models struggle with larger RPM: Claude and Gemini drop to trivial accuracy (around $12\%$) on **REL**-A3 $30 \times 30$; while GPT-5.2's accuracy drops by nearly $40\%$. On **REL**-A4, this trend is even more pronounced, with only GPT-5.2 achieving non-trivial accuracy ($21\%$) on $9 \times 9$ inputs. All three models fail on larger inputs.

Our RPT results show the opposite trend: **REL**-A5, **REL**-A6, and **REL**-A7 have, by design, a higher RC (4-6) on small inputs than the RPM tasks (1-3), but unlike with **REL**-A3 and **REL**-A4, their RC does not increase with the size of the input. As such, we find that more data, i.e. larger inputs, result in better model performance on **REL**-A5 and **REL**-A6 across all three models (for 56-64% to 77-87% on REL-A5 and from 41-50% to 53-64% on REL-A6). Only **REL**-A7 with its high initial RC (6) remains unsolvable for all models tested here, irrespective of input size, with an average accuracy of around $12\%$.

### 5.2. Biology Tasks

**Model evaluation.** An example is scored as correct only if the model both correctly detects the presence or absence of homoplasy and if presence, exactly identifies the set of homoplastic taxa. Failure to detect homoplasy or incorrect identification of taxa is counted as an incorrect prediction.

**Results.** Increasing the number of homoplastic taxa leads to a sharp decrease in model performance from 35% for $N_{ht} = 4$ to 1% for $N_{ht} = 25$ when averaged across models (Fig. 6). To assess whether this effect could be explained by alternative explanations we examined four other factors: motif ratio (the ratio of motif length to sequence length), sequence length, average pairwise distance between homoplastic taxa, and prompt length (Fig. 7). Increasing the motif ratio from 10-12.5% to 25-30% increased performance from 12.6% to 25.1%. Increasing the sequence length from 500 to 900 increased performance from 17.8% to 19.6%. Increasing distance from 5-6 to greater than 15 decreased performance from 10.2% to 3.2%. While these factors influence performance, none exhibit a comparably large effect across their full range (Fig. A.1).

To quantify the independent contribution of each factor, we fit separate multivariate regression models for each LLM and measured the unique share of explainable variance contributed by each variable. Across all three models, RC explains the largest share of explainable variance: $24\%$ of explainable variance for Claude, $32\%$ for Gemini, and $44\%$ for GPT. In contrast, the next strongest factor, motif ratio for Claude, prompt length for Gemini, and distance between taxa for GPT, explains $7\%$, $17\%$, and $6\%$ of explainable variance for Claude, Gemini, and GPT, respectively (Fig. 7). These results indicate that RC is a dominant driver of model performance, and that the observed performance degrada-

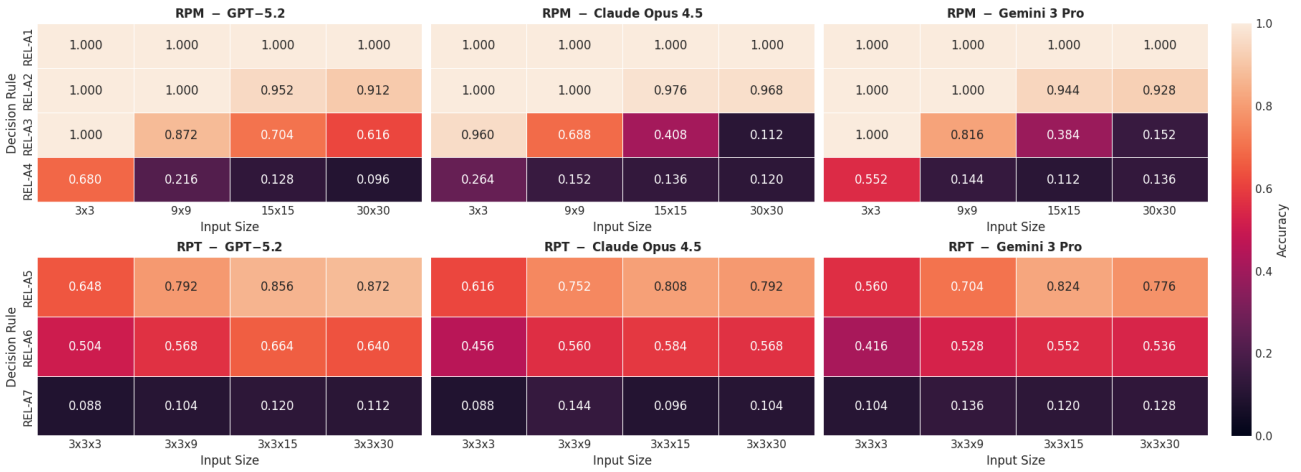


*Figure 5.* Model performance on **REL**-A tasks. RPMs at the top, with RPTs below. The models are given 8 answer choices, so trivial accuracy is 12.5%. All three model perform well on tasks with low RC (**REL**-A1 and **REL**-A2, top two rows), but struggle once RC increases: on **REL**-A3 and **REL**-A4, where RC increases with input size, performance drops by as much as 80%. RPTs (**REL**-A5, **REL**-A6, and **REL**-A7), which always have a higher RC, independent of input size, are challenging to impossible for all three models.

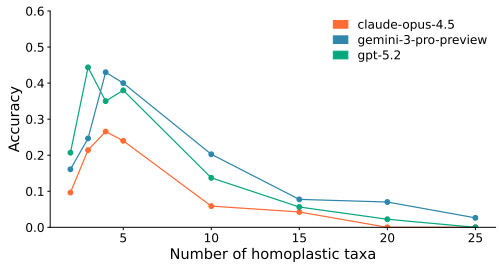


*Figure 6.* Performance decreases with increasing RC controlled by increasing the number of homoplastic taxa increases in **REL**-B.

tion in **REL**-B task cannot be explained as a direct consequence of other factors.

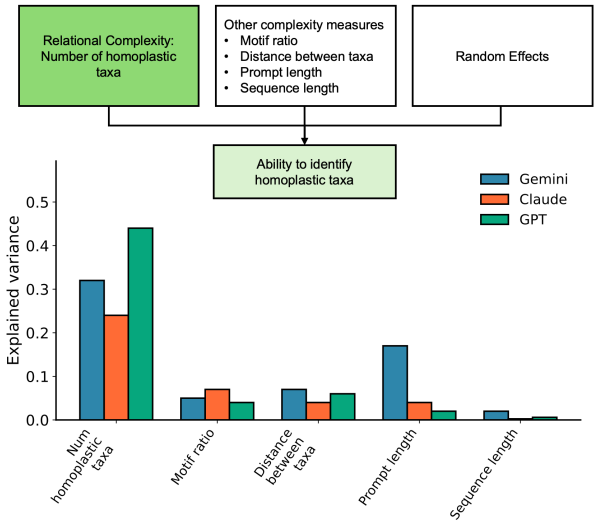


*Figure 7.* **Top:** Schema of variables in multivariate regression. **Bottom:** Explained variance of performance on **REL**-B across five measures of complexity. RC, which is number of homoplastic taxa, explains the most variance.

### 5.3. Chemistry Tasks

**Model evaluation.** All molecules are provided to the models as canonicalized SMILES. Model responses are evaluated by canonicalizing both predicted and ground-truth SMILES strings and comparing the canonical forms for exact match, ensuring that chemically equivalent SMILES representations are treated as correct. Example prompts of the three tasks are provided in Appendix D.3.

**Results.** With **REL**-C, we find that across our chemistry tasks, as RC increases, the task completion rate decreases (Fig. 8). **REL**-C1 has the highest average task completion rate at 65.7%, compared to **REL**-C2, which has an average task completion rate of 38.1%. While these two tasks share the same RC, OC of **REL**-C2 is higher than that of **REL**-C1. Finally, the task with the lowest task-completion rate is **REL**-C3 at 26.0%.

Both **REL**-C1 and **REL**-C2 have fixed RC with RC=2 across $N_{\text{molecules}}$. We observe that **REL**-C1 increases in accuracy from 56.0% at 5 molecules to 71.0% at 50 molecules.

In **REL**-C2, the task completion rate remains relatively stable for averaging at $39.2\% \pm 2.6\%$ for $5 \leq N_{\text{molecules}} < 50$ (Fig. 9L). This affirms that the number of entities does not affect performance until $N_{\text{molecules}}$ becomes very large. OC drives the decreased task completion rate between **REL**-C1 and **REL**-C2, as determining the MCS is more challenging than determining if molecules share the same molecular formulae. The OC of determining MCS also increases as the number of atoms in the input molecules increases, which is associated with a drop in performance (Fig. 9R).

Finally, for **REL**-C3, RC increases with both $N_{\text{isomers}}$ and $N_{\text{missing}} = N_{\text{isomers}} - N_{\text{observed}}$. We observe that increasing both sources of RC decreases performance across all three

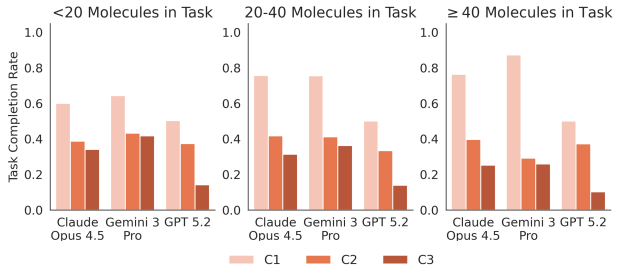


*Figure 8.* Task completion rate decreases as from C1 (RC=2 and OC easy) to C2 (RC=2 and OC medium) to C3 (RC=$N_{\text{isomers}}$, $N_{\text{observed}}$ and OC hard). This observation holds across different number of molecules in the task.

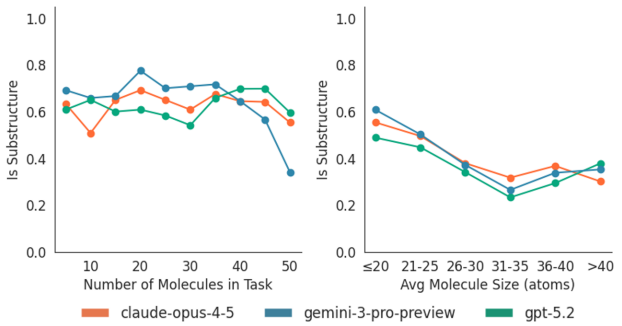


*Figure 9.* Task completion rate on **REL**-C2 evaluated with Is Substructure. For this task, RC is fixed at 2 in **Left:** increasing $N_{\text{molecules}}$ does not have an effect on performance until $N_{\text{molecules}} = 50$, in **Right:** increasing the molecule size increases OC and leads to decreased IsSubstructure rate.

models, with recall decreasing from 30.0% for $N_{\text{isomers}} = 5$ to 21.2% for $N_{\text{isomers}} = 50$ (Fig. 10L).

### 5.4. Sensitivity to Additional Hyperparameters

**Test-Time Compute.** We analyze how test-time compute affects performance on **REL**-A and **REL**-C. In Table A.5 we show results for REL-A4 and REL-A5 inputs where models are given a maximum token threshold of 4,096, 8,192, and 16,384 tokens. This tends to increase accuracy by 2-3% in accuracy, but does not close the gap to performance on tasks with lower RC. In Fig. A.4 we also show results for **REL**-C1,C2,C3 where models are again given 4,096, 8,192, and 16,384 tokens. On average, we observe a 0.4% change on average in task completion rate across the three models. With increased test-time compute, we still observe that higher RC leads to worse performance.

**In-Context Learning.** We run a variation of the task with one-shot in-context learning for 10% of our tasks in **REL**-C. At $N_{\text{molecules}} < 20$ we see that in-context learning leads to a boost in performance, however relationships between tasks remain unchanged with C1 at 77.0% (+6.6%), followed by C2 at 43.3% (+3.4%), and finally C3 at 32.7% (+6.0%). While in context learning improves performance, we observe the same outcome where as RC increases between

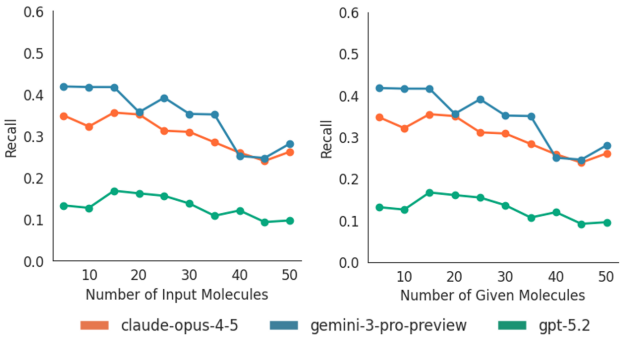


*Figure 10.* Task completion rate on **REL**-C3 evaluated with recall, RC depends on both $N_{\text{observed}}$ and $N_{\text{missing}}$. **Left:** As RC increases with $N_{\text{observed}}$ recall decreases. **Right:** As RC increases with $N_{\text{missing}}$ recall decreases.

questions, task completion rate decreases. We provide additional results in Appendix C.

## 6. Discussion

Across **REL**, our experiments reveal a consistent bottleneck: model performance tracks RC more reliably than traditional proxies such as input size or the number of entities. In **REL**-A, models solve low-RC RPM rules nearly perfectly, but accuracy collapses once the governing rule requires higher-arity integration. In **REL**-B, RC, controlled by the number of homoplastic taxa, explains most of the variance in performance degradation. Finally, in chemistry, increasing RC across **REL**-C1, **REL**-C2, and **REL**-C3 induces a uniform decline in task completion rates.

These findings have two implications. First, many benchmark improvements that appear to reflect better reasoning may instead arise from gains along axes orthogonal to relational integration, and re-evaluating benchmarks through the lens of RC may therefore yield more diagnostic comparisons across models and inference settings. Second, the observed failure regime appears persistent: allocating additional test-time thinking provides limited benefit on high-RC instances, and several tasks remain effectively unsolved across all evaluated models.

Our study has several limitations: multiple-choice evaluation may obscure finer-grained reasoning failures, context-length constraints lead to invalid responses in some settings, and our tasks remain relatively synthetic. Addressing these limitations is an important direction for future work. Going forward, we aim to expand **REL** to more naturalistic relational settings and to explore approaches to improve target higher RC reasoning. With **REL**, we provide a framework for evaluating model performance based on their ability to reliably compose many relations simultaneously.

## Impact Statement

This paper aims to advance machine learning by characterizing how large language model (LLM) performance varies with relational complexity, situations where solving a problem requires representing and manipulating multiple relations simultaneously. As LLMs are increasingly deployed in real-world settings, understanding systematic failure modes is important for safer and more appropriate use. Our results suggest that increasing relational complexity can lead to substantial performance degradation, which may inform practitioners about when additional safeguards, alternative methods, or human oversight are warranted, especially in higher-stakes applications.

## References


Acharya, A., Yadav, M., Nagpure, M., Kumaresan, S., and Guchhait, S. K. Molecular medicinal insights into scaffold hopping-based drug discovery success. *Drug Discovery Today*, 29(1):103845, 2024. ISSN 1359-6446. doi: https://doi.org/10.1016/j.drudis.2023.103845. URL https://www.sciencedirect.com/science/article/pii/S1359644623003616.

Alexander, P. A., Dumas, D., Grossnickle, E. M., List, A., and Firetto, C. M. Measuring relational reasoning. *The Journal of Experimental Education*, 84(1):119–151, 2016.

Austin, J., Odena, A., Nye, M., Bosma, M., Michalewski, H., Dohan, D., Jiang, E., Cai, C., Terry, M., Le, Q., et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Bemis, G. W. and Murcko, M. A. The properties of known drugs. 1. molecular frameworks. *Journal of medicinal chemistry*, 39(15):2887–2893, 1996.

Bianchi, O., Koretsky, M. J., Willey, M., Alvarado, C. X., Nayak, T., Asija, A., Kuznetsov, N., Nalls, M. A., Faghri, F., and Khashabi, D. Hidden in the haystack: Smaller needles are more difficult for llms to find. 2025. URL https://arxiv.org/abs/2505.18148.

Bouhaddou, M., Reuschl, A.-K., Polacco, B. J., Thorne, L. G., Ummadi, M. R., Ye, C., Rosales, R., Pelin, A., Batra, J., Jang, G. M., Xu, J., Moen, J. M., Richards, A. L., Zhou, Y., Harjai, B., Stevenson, E., Rojc, A., Ragazzini, R., Whelan, M. V. X., Furnon, W., De Lorenzo, G., Cowton, V., Syed, A. M., Ciling, A., Deutsch, N., Pirak, D., Dowgier, G., Mesner, D., Turner, J. L., McGovern, B. L., Rodriguez, M. L., Leiva-Rebollo, R., Dunham, A. S., Zhong, X., Eckhardt, M., Fossati, A., Liotta, N. F., Kehrer, T., Cupic, A., Rutkowska, M., Mena, I., Aslam, S., Hoffert, A., Foussard, H., Olwal, C. O., Huang, W., Zwaka, T., Pham, J., Lyons, M., Donohue, L., Griffin, A., Nugent, R., Holden, K., Deans, R., Aviles, P., Lopez-Martin, J. A., Jimeno, J. M., Obernier, K., Fabius, J. M., Soucheray, M., Hüttenhain, R., Jungreis, I., Kellis, M., Echeverria, I., Verba, K., Bonfanti, P., Beltrao, P., Sharan, R., Doudna, J. A., Martinez-Sobrido, L., Patel, A. H., Palmarini, M., Miorin, L., White, K., Swaney, D. L., Garcia-Sastre, A., Jolly, C., Zuliani-Alvarez, L., Towers, G. J., and Krogan, N. J. SARS-CoV-2 variants evolve convergent strategies to remodel the host response. *Cell*, 186(21):4597–4614.e26, October 2023.

Brouwers, S. A., Van de Vijver, F. J., and Van Hemert, D. A. Variation in raven's progressive matrices scores across time and place. *Learning and Individual Differences*, 19 (3):330–338, 2009.

Burke, H. R. Raven's progressive matrices: Validity, reliability, and norms. *The Journal of Psychology*, 82(2): 253–257, 1972.

Camposampiero, G., Hersche, M., Wattenhofer, R., Sebastian, A., and Rahimi, A. Can large reasoning models do analogical reasoning under perceptual uncertainty? *arXiv preprint arXiv:2503.11207*, 2025a.

Camposampiero, G., Hersche, M., Wattenhofer, R., Sebastian, A., and Rahimi, A. I-raven-x: Benchmarking generalization and robustness of analogical and mathematical reasoning in large language and reasoning models. *arXiv preprint arXiv:2510.17496*, 2025b.

Carpenter, P. A., Just, M. A., and Shell, P. What one intelligence test measures: a theoretical account of the processing in the raven progressive matrices test. *Psychological review*, 97(3):404, 1990.

Chen, M. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Chen, X., Wang, T., Guo, T., Guo, K., Zhou, J., Li, H., Song, Z., Gao, X., and Zhang, X. Unveiling the power of language models in chemical research question answering. *Communications Chemistry*, 8(1):4, 2025.

Cheng, W., Song, Z., Zhang, Y., et al. Dnalongbench: A benchmark suite for long-range dna prediction tasks. *Nature Communications*, 16:10108, 2025. doi: 10.1038/s41467-025-65077-4.

Clark, P., Cowhey, I., Etzioni, O., Khot, T., Sabharwal, A., Schoenick, C., and Tafjord, O. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Crispell, J., Balaz, D., and Gordon, S. V. HomoplasyFinder: a simple tool to identify homoplasies on a phylogeny. *Microb. Genom.*, 5(1), January 2019.

Crone, E. A., Wendelken, C., Van Leijenhorst, L., Honomichl, R. D., Christoff, K., and Bunge, S. A. Neurocognitive development of relational reasoning. *Developmental science*, 12(1):55–66, 2009.

Deveci, I. E. and Ataman, D. The ouroboros of benchmarking: Reasoning evaluation in an era of saturation. *arXiv preprint arXiv:2511.01365*, 2025.

Dumas, D., Alexander, P. A., and Grossnickle, E. M. Relational reasoning and its manifestations in the educational context: A systematic review of the literature. *Educational Psychology Review*, 25(3):391–427, 2013.

Dumas, D., Alexander, P. A., Baker, L. M., Jablansky, S., and Dunbar, K. N. Relational reasoning in medical education: Patterns in discourse and diagnosis. *Journal of Educational Psychology*, 106(4):1021, 2014.

Edge, D., Trinh, H., Cheng, N., Bradley, J., Chao, A., Mody, A., Truitt, S., Metropolitansky, D., Ness, R. O., and Larson, J. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*, 2024.

Fang, Y., Liang, X., Zhang, N., Liu, K., Huang, R., Chen, Z., Fan, X., and Chen, H. Mol-instructions: A large-scale biomolecular instruction dataset for large language models. *arXiv preprint arXiv:2306.08018*, 2023.

Fatemi, B., Halcrow, J., and Perozzi, B. Talk like a graph: Encoding graphs for large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=IuXR1CCrSi.

Gao, Z., Wang, H., Tan, C., Xu, C., Liu, M., Hu, B., Chao, L., Zhang, X., and Li, S. Z. Pfmbench: Protein foundation model benchmark, 2025. URL https://arxiv.org/abs/2506.14796.

Gerlinger, M., Rowan, A. J., Horswell, S., Larkin, J., Endesfelder, D., Gronroos, E., Martinez, P., Matthews, N., Stewart, A., Tarpey, P., Varela, I., Phillimore, B., Begum, S., McDonald, N. Q., Butler, A., Jones, D., Raine, K., Latimer, C., Santos, C. R., Nohadani, M., Eklund, A. C., Spencer-Dene, B., Clark, G., Pickering, L., Stamp, G., Gore, M., Szallasi, Z., Downward, J., Futreal, P. A., and Swanton, C. Intratumor heterogeneity and branched evolution revealed by multiregion sequencing. *New England Journal of Medicine*, 366(10):883–892, 2012. doi: 10.1056/NEJMoa1113205. URL https://www.nejm.org/doi/full/10.1056/NEJMoa1113205.

Geva, M., Khashabi, D., Segal, E., Khot, T., Roth, D., and Berant, J. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361, 2021.

Guha, R. On exploring structure–activity relationships. *In silico models for drug discovery*, pp. 81–94, 2013.

Guo, T., Nan, B., Liang, Z., Guo, Z., Chawla, N., Wiest, O., Zhang, X., et al. What can large language models do in chemistry? a comprehensive benchmark on eight tasks. *Advances in Neural Information Processing Systems*, 36:59662–59688, 2023.

Hajduk, P. J., Galloway, W. R., and Spring, D. R. A question of library design. *Nature*, 470(7332):42–43, 2011.

Halford, G. S., Wilson, W. H., and Phillips, S. Processing capacity defined by relational complexity: Implications for comparative, developmental, and cognitive psychology. *Behavioral and brain sciences*, 21(6):803–831, 1998.

He, X., Tian, Y., Sun, Y., Chawla, N., Laurent, T., LeCun, Y., Bresson, X., and Hooi, B. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *Advances in Neural Information Processing Systems*, 37:132876–132907, 2024.

Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., Song, D., and Steinhardt, J. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Hersche, M., Camposampiero, G., Wattenhofer, R., Sebastian, A., and Rahimi, A. Towards Learning to Reason: Comparing LLMs with Neuro-Symbolic on Arithmetic Relations in Abstract Reasoning. In *AAAI Workshop on Neural Reasoning and Mathematical Discovery – An Interdisciplinary Two-Way Street (NEURMAD)*, 2025.

Ho, X., Nguyen, A.-K. D., Sugawara, S., and Aizawa, A. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. *arXiv preprint arXiv:2011.01060*, 2020.

Huang, Y., Zhang, R., He, X., Zhi, X., Wang, H., Li, X., Xu, F., Liu, D., Liang, H., Li, Y., et al. Chemeval: a comprehensive multi-level chemical evaluation for large language models. *arXiv preprint arXiv:2409.13989*, 2024.

Jang, Y., Kim, J., and Ahn, S. Improving chemical understanding of LLMs via SMILES parsing. In Christodoulopoulos, C., Chakraborty, T., Rose, C., and Peng, V. (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp.

15683–15698, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.791. URL https://aclanthology.org/2025.emnlp-main.791/.

John and Raven, J. Raven progressive matrices. In *Handbook of nonverbal assessment*, pp. 223–237. Springer, 2003.

Li, H., Cao, H., Feng, B., Shao, Y., Tang, X., Yan, Z., Yuan, L., Tian, Y., and Li, Y. Beyond chemical qa: Evaluating llm's chemical reasoning with modular chemical operations. *arXiv preprint arXiv:2505.21318*, 2025.

Li, M., Zhao, Y., Yu, B., Song, F., Li, H., Yu, H., Li, Z., Huang, F., and Li, Y. Api-bank: A comprehensive benchmark for tool-augmented llms. *arXiv preprint arXiv:2304.08244*, 2023.

Li, M., Miao, S., and Li, P. Simple is effective: The roles of graphs and large language models in knowledge-graph-based retrieval-augmented generation. *arXiv preprint arXiv:2410.20724*, 2024.

Liévin, V., Hother, C. E., Motzfeldt, A. G., and Winther, O. Can large language models reason about medical questions? *Patterns*, 5(3), 2024.

Liu, A., Prior, H., Balasubramaniam, G., Moroshko, R., Zait, A., Labzovsky, I., Karmon, D., Dasgupta, I., Stachenfeld, K., and Marino, K. Recoglab: a framework testing relational reasoning & cognitive hypotheses on LLMs. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=yORSk4Ycsa.

Liu, J., Cui, L., Liu, H., Huang, D., Wang, Y., and Zhang, Y. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. *arXiv preprint arXiv:2007.08124*, 2020.

Liu, X., Ouyang, S., Zhong, X., Han, J., and Zhao, H. FG-Bench: A dataset and benchmark for molecular property reasoning at functional group-level in large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2025b. URL https://openreview.net/forum?id=VIsPHMMiW8.

Luo, L., Li, Y.-F., Haffari, G., and Pan, S. Reasoning on graphs: Faithful and interpretable large language model reasoning. *arXiv preprint arXiv:2310.01061*, 2023.

Markov, P. V., Ghafari, M., Beer, M., Lythgoe, K., Simmonds, P., Stilianakis, N. I., and Katzourakis, A. The evolution of sars-cov-2. *Nature Reviews Microbiology*, 21(6): 361–379, jun 2023. doi: 10.1038/s41579-023-00878-2. URL https://doi.org/10.1038/s41579-023-00878-2.

McGranahan, N. and Swanton, C. Clonal heterogeneity and tumor evolution: Past, present, and the future. *Cell*, 168 (4):613–628, February 2017.

McKay, B. D., Yirik, M. A., and Steinbeck, C. Surge: a fast open-source chemical graph generator. *Journal of cheminformatics*, 14(1):24, 2022.

Mendez, D., Gaulton, A., Bento, A. P., Chambers, J., De Veij, M., Félix, E., Magariños, M. P., Mosquera, J. F., Mutowo, P., Nowotka, M., et al. Chembl: towards direct deposition of bioassay data. *Nucleic acids research*, 47 (D1):D930–D940, 2019.

Mills, C. J., Ablard, K. E., and Brody, L. E. The raven's progressive matrices: Its usefulness for identifying gifted/talented students. *Roeper Review*, 15(3): 183–186, 1993.

Mirza, A., Alampara, N., Kunchapu, S., Ríos-García, M., Emoekabu, B., Krishnan, A., Gupta, T., Schilling-Wilhelmi, M., Okereke, M., Aneesh, A., et al. A framework for evaluating the chemical knowledge and reasoning abilities of large language models against the expertise of chemists. *Nature Chemistry*, pp. 1–8, 2025.

Notin, P., Kollasch, A. W., Ritter, D., Niekerk, L. V., Paul, S., Spinner, H., Rollins, N. J., Shaw, A., Orenbuch, R., Weitzman, R., Frazer, J., Dias, M., Franceschi, D., Gal, Y., and Marks, D. S. Proteingym: Large-scale benchmarks for protein fitness prediction and design. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=URoZHqAohf.

Qin, Y., Liang, S., Ye, Y., Zhu, K., Yan, L., Lu, Y., Lin, Y., Cong, X., Tang, X., Qian, B., Zhao, S., Hong, L., Tian, R., Xie, R., Zhou, J., Gerstein, M., dahai li, Liu, Z., and Sun, M. ToolLLM: Facilitating large language models to master 16000+ real-world APIs. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=dHng2O0Jjr.

Rao, R., Bhattacharya, N., Thomas, N., Duan, Y., Chen, X., Canny, J., Abbeel, P., and Song, Y. S. Evaluating protein transfer learning with tape, 2019. URL https://arxiv.org/abs/1906.08230.

Rong, D., Chen, Z., Jia, Q., Zhang, K., Lu, H., Zhai, G., and Liu, N. Liveproteinbench: A contamination-free benchmark for assessing models' specialized capabilities in protein science. 2025. URL https://arxiv.org/abs/2512.22257.

Runcie, N. T., Deane, C. M., and Imrie, F. Assessing the chemical intelligence of large language models. *arXiv preprint arXiv:2505.07735*, 2025. doi: 10.48550/arXiv.2505.07735.

Stern, D. L. The genetic causes of convergent evolution. *Nature Reviews Genetics*, 14(11):751–764, 2013. doi: 10.1038/nrg3483. URL https://doi.org/10.1038/nrg3483.

Storz, J. Causes of molecular convergence and parallelism in protein evolution. *Nature Reviews Genetics*, 17(4): 239–250, apr 2016. doi: 10.1038/nrg.2016.11. URL https://doi.org/10.1038/nrg.2016.11.

Tafjord, O., Dalvi, B., and Clark, P. Proofwriter: Generating implications, proofs, and abductive statements over natural language. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 3621–3634, 2021.

Tang, J., Zhang, Q., Li, Y., Chen, N., and Li, J. Grapharena: Evaluating and exploring large language models on graph computation. *arXiv preprint arXiv:2407.00379*, 2024.

Trivedi, H., Balasubramanian, N., Khot, T., and Sabharwal, A. Musique: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022.

Wake, D. B. Homoplasy: The result of natural selection, or evidence of design limitations? *The American Naturalist*, 138(3):543–567, sep 1991. doi: 10.1086/285234. URL https://doi.org/10.1086/285234.

Wake, D. B., Wake, M. H., and Specht, C. D. Homoplasy: from detecting pattern to determining process and mechanism of evolution. *Science*, 331(6020):1032–1035, 2011.

Wang, H., Feng, S., He, T., Tan, Z., Han, X., and Tsvetkov, Y. Can language models solve graph problems in natural language? *Advances in Neural Information Processing Systems*, 36:30840–30861, 2023a.

Wang, X., Hu, Z., Lu, P., Zhu, Y., Zhang, J., Subramaniam, S., Loomba, A. R., Zhang, S., Sun, Y., and Wang, W. Scibench: Evaluating college-level scientific problem-solving abilities of large language models. *arXiv preprint arXiv:2307.10635*, 2023b.

Williams, J. E. and McCord, D. M. Equivalence of standard and computerized versions of the raven progressive matrices test. *Computers in Human Behavior*, 22(5):791–800, 2006.

Wu, Q., Chen, Z., Corcoran, W., Sra, M., and Singh, A. GraphEval36K: Benchmarking coding and reasoning capabilities of large language models on graph datasets. In Chiruzzo, L., Ritter, A., and Wang, L. (eds.), *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 8095–8117, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-195-7. doi: 10.18653/v1/2025.findings-naacl.452. URL https://aclanthology.org/2025.findings-naacl.452/.

Xu, M., Zhang, Z., Lu, J., Zhu, Z., Zhang, Y., Ma, C., Liu, R., and Tang, J. Peer: A comprehensive and multi-task benchmark for protein sequence understanding, 2022. URL https://arxiv.org/abs/2206.02096.

Yang, Z., Qi, P., Zhang, S., Bengio, Y., Cohen, W., Salakhutdinov, R., and Manning, C. D. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In Riloff, E., Chiang, D., Hockenmaier, J., and Tsujii, J. (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

Ye, F., Zheng, Z., Xue, D., Shen, Y., Wang, L., Ma, Y., Wang, Y., Wang, X., Zhou, X., and Gu, Q. Proteinbench: A holistic evaluation of protein foundation models, 2024. URL https://arxiv.org/abs/2409.06744.

Zhang, Y., Wang, H., Feng, S., Tan, Z., Han, X., He, T., and Tsvetkov, Y. Can llm graph reasoning generalize beyond pattern memorization? *arXiv preprint arXiv:2406.15992*, 2024.

Zhou, T., Fu, D., Soltanolkotabi, M., Jia, R., and Sharan, V. Fone: Precise single-token number embeddings via fourier features. *arXiv preprint arXiv:2502.09741*, 2025.

Zhu, X., Xie, Y., Liu, Y., Li, Y., and Hu, W. Knowledge graph-guided retrieval augmented generation. *arXiv preprint arXiv:2502.06864*, 2025.

# A. Details on the Construction of REL Tasks

## A.1. REL-Algebra

For each problem size and ground rule, we generate 125 distinct RPMs/ RPTs.

- For **REL**-A1, we sample an integer value randomly from a predefined domain.
- For **REL**-A2, we randomly choose "plus" or "minus" for the progression and then sample an integer increment, as well as initial values for each row uniformly from a predefined domain.
- For **REL**-A2, we randomly choose "plus" or "minus" for the progression and then sample an integer increment, as well as initial values for each row uniformly from a predefined domain.
- For **REL**-A4, we sample "plus" or "minus" $n-1$ times and then populate the first $n-1$ columns with integers from a predefined domain, which determines the last column.
- The RPTs for **REL**-A5 and **REL**-A6 are both generated using randomly sampled values for the first $3 \times 3$ values of the RPT, and the moving average rule then determines the rest of the tensor.
- Finally, generating **REL**-A7 RPTs requires a more involved algorithm, which we elaborate on below.

**REL-A7.**

**Problem Definition.** Given parameters:

- Grid size: $n \times n$ (typically $n = 3$)
- Number of slices: $K$ (depth of tensor)
- Maximum value: maxval
- Prime modulus: $p$

Generate a 3D tensor $A \in \mathbb{Z}p^{n \times n \times K}$ such that for each slice $k \in \{0, 1, \ldots, K-1\}$ and each cell $(i, j)$, the value $Ai, j, k$ satisfies the neighborhood sum constraint modulo $p$ using 4-connected neighbors (cardinal directions).

**Neighborhood Definition.** For a cell at position $(i, j)$ in slice $k$, define the 4-connected neighborhood set $\mathcal{N}_{i,j}$:

$$\mathcal{N}_{i,j} = \{(i-1 \bmod n, j), (i+1 \bmod n, j), (i, j-1 \bmod n), (i, j+1 \bmod n)\} \tag{2}$$

**Mathematical Formulation.** For each slice $k \in \{0, 1, \ldots, K-1\}$ and cell $(i, j)$, the constraint is:

$$A_{i,j,k} \equiv \sum_{(i',j') \in \mathcal{N}i,j} A_{i',j',k} \pmod{p} \tag{3}$$

**Algorithm 1** Self-Consistent NeighborhoodSum RPT Generation

**Require:** $n, K, \text{maxval}, p$
**Ensure:** Tensor $A \in \mathbb{Z}_p^{n \times n \times K}$ where every cell satisfies the neighborhood constraint, missing cell position $(k, i, j)$, target value $t$, and answer candidates

Initialize $A$ randomly: $A_{i,j,k} \leftarrow \text{RandomInt}(0, \text{maxval})$ for all $i, j, k$
converged $\leftarrow$ False
**while** not converged **do**
  converged $\leftarrow$ True
  **for** $k = 0$ to $K - 1$ **do**
    **for** $i = 0$ to $n - 1$ **do**
      **for** $j = 0$ to $n - 1$ **do**
        sum $\leftarrow 0$
        **for** $(i', j') \in \mathcal{N}_{i,j}$ **do**
          sum $\leftarrow$ sum $+ A_{i',j',k}$
        **end for**
        new_val $\leftarrow$ sum $\bmod p$
        **if** $A_{i,j,k} \neq$ new_val **then**
          $A_{i,j,k} \leftarrow$ new_val
          converged $\leftarrow$ False
        **end if**
      **end for**
    **end for**
  **end for**
**end while**
**Select missing cell**
$(k, i, j^*) \leftarrow \text{RandomChoice}(\{(k, i, j) : k \in \{0, \ldots, K-1\}, i, j \in \{0, \ldots, n-1\}\})$
$t^* \leftarrow A_{k,i,j^*}$
**Generate answer candidates**
candidates $\leftarrow [t^*]$ {Correct answer}
**while** |candidates| $< 8$ **do**
  $d \leftarrow \text{RandomDistractor}()$ {Generate distractor value}
  **if** $d \notin$ candidates **then**
    candidates.append$(d)$
  **end if**
**end while**
candidates $\leftarrow$ Shuffle(candidates)
**Return** $(A, (k, i, j), t^*, \text{candidates})$

### A.2. REL-Biology

Each dataset instance is generated as follows:

1. **Sample a random tree.** Draw a random Newick tree with $n_{\text{leaves}}$ taxa.
2. **Simulate a baseline alignment.** Using Pyvolve, simulate a nucleotide alignment of length $l_{\text{seq}}$ under a standard substitution model.
3. **Inject tree-aware convergent blocks.** Inject a motif of length $l_{\text{motif}}$ by enforcing a shared motif across taxa that are distant on the tree:
   - Select $n_{ht}$ leaves whose pairwise *topological distance* (the number of edges along the unique path between two leaves) is at least 3.
   - For a randomly chosen contiguous block of $l_{\text{motif}}$ columns, overwrite the nucleotides for the selected taxa with the same base (or motif), inducing a structured convergence signal spanning multiple columns.

We generate datasets starting from a baseline configuration with $n_{\text{leaves}} = 50$, $l_{\text{seq}} = 1000$, $l_{\text{motif}} = 50$, and $n_{ht} = 2$, and then vary one parameter at a time. Specifically, we consider $n_{ht} \in \{2, 3, 4, 5, 10, 15, 20, 25\}$, $n_{\text{leaves}} \in \{20, 30, 40, 100, 1000\}$, $l_{\text{seq}} \in \{200, 300, 500, 1000, 2000\}$, and $l_{\text{motif}} \in \{3, 4, 5, 30, 40, 50\}$, we also resulting in a total of 2,600 questions. Across the three evaluated LLMs, this yields 7,800 API calls.

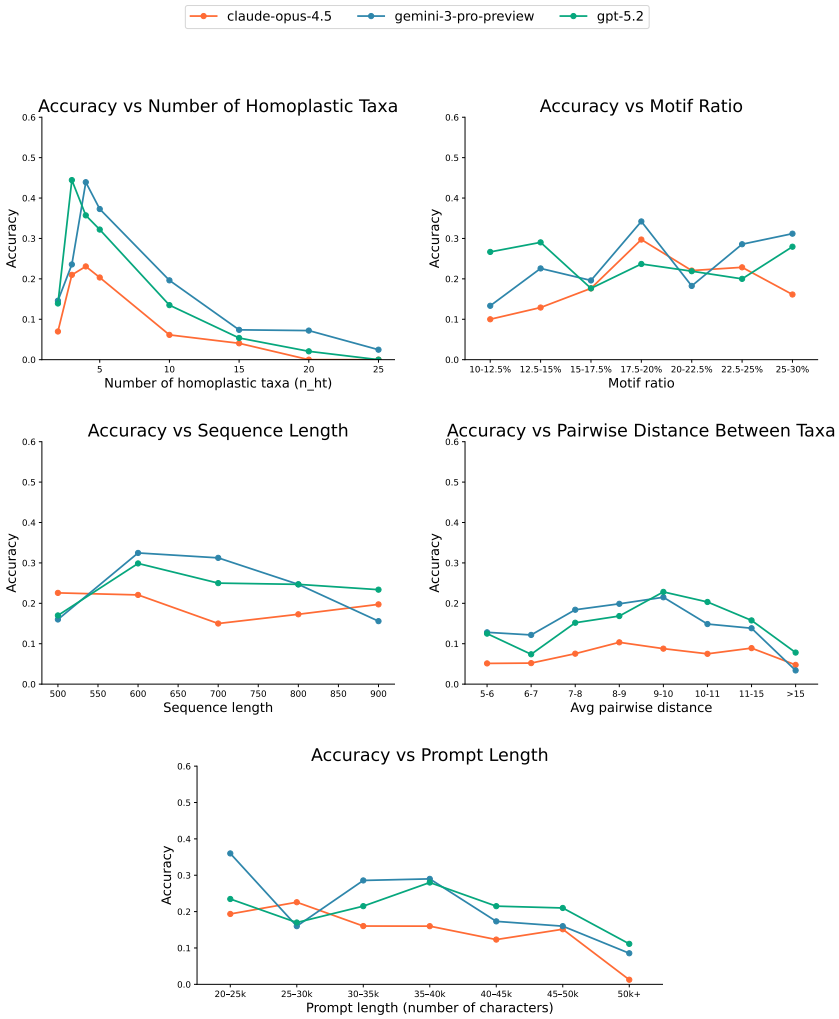


*Figure A.1.* Accuracy as a function of the number of homoplastic taxa, motif ratio, sequence length, average pairwise distance between taxa, and prompt length for **REL**-B.

## A.3. REL-Chemistry

### A.3.1. **REL**-C1.

For the candidate formulas spanning $C_{3-9}$ with various heteroatoms (O, N, S, F, Cl, Br) and degrees of unsaturation, we generate isomers using the Surge structure enumeration tool (McKay et al., 2022). After generation, we filter to retain formulas with 5-100 isomers to ensure both tractability and sufficient sampling diversity.

*Table A.1.* Details for **REL**-C1. Below double bond equivalent (DBE) measures the degree of unsaturation in the formula.

| $N_{\text{molecules}}$ | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 |
|---|---|---|---|---|---|---|---|---|---|---|
| # Questions | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| Avg. Atoms | 5.75 | 5.78 | 5.94 | 6.10 | 6.20 | 6.10 | 6.34 | 6.21 | 6.15 | 6.32 |
| Avg. DBE | 4.31 | 4.54 | 4.61 | 4.68 | 4.78 | 4.80 | 4.79 | 5.35 | 5.31 | 5.47 |

### A.3.2. **REL**-C2.

To select molecules with real-world relevance, molecules in this task are sampled from phase 1 to 4 drug molecules in ChEMBL (Mendez et al., 2019). We first construct a diverse molecular bank by filtering molecules to contain 15-60 heavy atoms and applying a greedy diversity selection algorithm that iteratively selects molecules maximally distant (by Tanimoto distance) from those already selected resulting in 9,035 molecules. For each instance at a given $N_{\text{molecules}}$, we randomly select a seed molecule and sample similar molecules from the similarity range [0.35, 0.90], ensuring structural relatedness while avoiding near-duplicates. We impose a minimum MCS size of 8 atoms, as lower thresholds lead to many trivial motifs, primarily consisting of benzene and toluene substructures. In total, we generated 1,016 questions with $N_{\text{molecules}} \in \{5, 10, 15, 20, 25, 30, 35, 40, 45, 50\}$ the number of questions for each number of molecules is provided in Table A.2.

*Table A.2.* **REL**-C2. Number of questions by number of molecules. The number of questions decreases with the number of molecules as it becomes more unlikely to sample a set of molecules with a largest common motif of at least 8 atoms.

| $N_{\text{molecules}}$ | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 |
|---|---|---|---|---|---|---|---|---|---|---|
| # Questions | 120 | 120 | 120 | 120 | 120 | 120 | 120 | 76 | 53 | 47 |
| Avg. atoms | 30.74 | 30.75 | 27.28 | 28.18 | 27.25 | 27.39 | 26.46 | 27.42 | 27.42 | 28.45 |
| Avg. elements | 3.32 | 3.34 | 3.29 | 3.34 | 3.30 | 3.25 | 3.14 | 3.02 | 2.96 | 2.83 |
| Avg. DBE | 24.55 | 24.59 | 22.11 | 22.55 | 22.09 | 22.12 | 21.79 | 23.08 | 23.27 | 24.08 |
| Avg. rings | 3.03 | 2.97 | 2.98 | 3.00 | 3.09 | 3.07 | 3.13 | 3.32 | 3.42 | 3.81 |

*Table A.3.* Most frequent molecular substructure motifs and their occurrence counts in **REL**-C2.

| Motif (SMILES) | Count |
|---|---|
| COc1ccccc1 | 180 |
| CCc1ccccc1 | 114 |
| O=[SH](=O)c1ccccc1 | 50 |
| CCOc1ccccc1 | 17 |
| CC12C=CC(=O)C=C1CCC1C2CCC2(C)C(C=O)CCC12 | 13 |
| CC(C)NCC(O)COc1ccccc1 | 9 |
| CCCCCCOC | 5 |
| CCCOc1ccccc1 | 5 |
| CC(O)C(O)C(O)CO | 4 |
| CCN(CC)CCCN1c2ccccc2Sc2ccccc21 | 4 |

### A.3.3. **REL**-C3.

For the missing isomer completion task, we leverage the same exhaustively enumerated constitutional isomer universes used in C1. For each instance at a given $N_{\text{molecules}}$, we select a molecular formula whose complete isomer universe has between 8 and 100 members. From this complete universe, we randomly sample $N_{\text{molecules}}$ as the "given" set presented to the model, with the remaining molecules constituting the ground-truth answer. The average number of isomers to be identified is 29. For this task, we define the task completion rate as the recall of correct isomers.

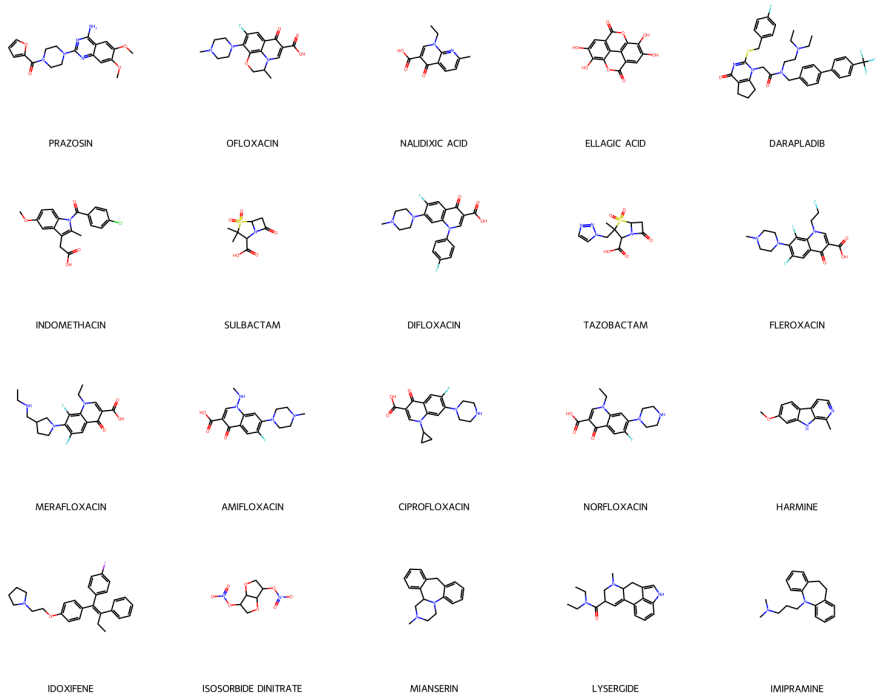


*Figure A.2.* Sample of 20 molecules used in **REL**-C2.

*Table A.4.* **REL**-C3. Number of missing isomers as a function of the number of molecules.

| $N_{\text{molecules}}$ | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 |
|---|---|---|---|---|---|---|---|---|---|---|
| # Questions | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| # Missing Isomers | 36.41 | 32.92 | 29.93 | 30.39 | 24.13 | 27.28 | 24.17 | 33.24 | 28.16 | 26.44 |
| Avg. atoms | 5.91 | 5.83 | 5.84 | 6.22 | 6.23 | 6.19 | 6.42 | 6.18 | 6.22 | 6.31 |
| Avg. DBE | 4.56 | 4.69 | 4.73 | 4.74 | 4.59 | 4.78 | 4.66 | 5.39 | 5.34 | 5.40 |

# B. Extended Background

## B.1. Relational reasoning in Algebra

Algebra is a prototypical domain for relational reasoning because its primitives are defined by relations, most centrally equality and equivalence, and progress is made by applying transformations that preserve those relations. In typical mathematics tasks (solving equations, simplifying expressions, proving identities), a solver must maintain multiple constraints at once (e.g., which quantities are bound together, which substitutions are valid, which invariants are preserved) while manipulating symbolic structure. This emphasis on structure-preserving transformation makes algebra a natural setting for analyzing difficulty through the number of simultaneously active "slots" or variables that must be integrated in a single reasoning step, as formalized by relational complexity theory.

Raven's Progressive Matrices (John & Raven, 2003; Burke, 1972; Mills et al., 1993) can be viewed as an abstract completion problem in exactly this sense: the missing entry is determined by a rule that relates other entries, and solving requires inducing and composing those relations across the matrix. In **REL**-A, we use an algebraic reframing of RPMs (Camposampiero et al., 2025b; Hersche et al., 2025) to control difficulty directly by controlling dependency: increasing the number of entries the missing value increases the arity of the governing relation and thus the relational complexity. Extending from matrices to Raven's Progressive Tensors (RPTs) further enlarges this design space, enabling local neighborhood rules (neighbor sums) whose relational bottleneck scales with the size of the dependency set.

### B.2. Relational reasoning in Biology

In biology, many inferences require relational comparisons across organisms against an evolutionary backdrop. A canonical example is convergent evolution where the same mutation (or short sequence motif) arises independently in distinct lineages, which can indicate functional constraint or shared selection pressures (Stern, 2013; Storz, 2016) (e.g., viral adaptation (Markov et al., 2023; Bouhaddou et al., 2023), recurrent cancer tumor drivers (Gerlinger et al., 2012; McGranahan & Swanton, 2017)). In phylogenetics, such repeated, independent changes are referred to as homoplasy. Detecting homoplasy requires two inputs: (i) a shared motif in the multiple sequence alignment (MSA), and (ii) a phylogenetic tree establishing that the shared state is not explained by a recent common ancestor. Operationally, a motif shared by a subset of taxa is homoplasic if those taxa are evolutionarily separated on the tree (e.g., occupy different clades) such that shared ancestry alone cannot explain the pattern (Crispell et al., 2019; Wake, 1991).

### B.3. Relational Reasoning in Chemistry

Many benchmarks evaluate LLMs on their ability to reason for questions that require chemistry domain knowledge, including ChemLLMBench (Guo et al., 2023), ChemEval (Huang et al., 2024), ChemBench (Mirza et al., 2025), and ScholarChemQA (Chen et al., 2025). Several benchmarks also evaluate the ability of models for SMILES comprehension. For example, Mol-Instructions (Fang et al., 2023), CleanMol (Jang et al., 2025), and ChemIQ (Runcie et al., 2025) evaluate the model ability for graph-level molecular comprehension. ChemCoTBench (Li et al., 2025) involves granular tasks with molecular editing of SMILES for property optimization and reaction prediction. FGBench (Liu et al., 2025b) focuses on functional group-level reasoning for molecular property prediction. Previous benchmarks focus primarily on individual molecules or, at most, reactant-product relationships in chemical reactions, limiting their ability to evaluate whether LLMs can reason across sets of molecules and infer higher-order relations.

Relational reasoning with the understanding of shared structure across molecules is a fundamental aspect of chemistry. Bemis & Murcko (1996) introduced a formal definition of molecular scaffolds to organize and compare large collections of drug-like molecules, influencing how chemical space is structured and explored in drug discovery. This scaffold-based view underpins scaffold hopping, a standard strategy for exploring new druggable regions of chemical space (Acharya et al., 2024). More broadly, identifying chemical relationships at scale is central to library design (Hajduk et al., 2011) and the analysis of structure–activity relationships (Guha, 2013), both of which aim to enable more systematic exploration of vast chemical spaces.

## C. Additional Experiments

### C.1. In-Context Learning

In Fig A.3 we how one-shot in-context learning changes the task completion rate across **REL**-C1,C2,C3.

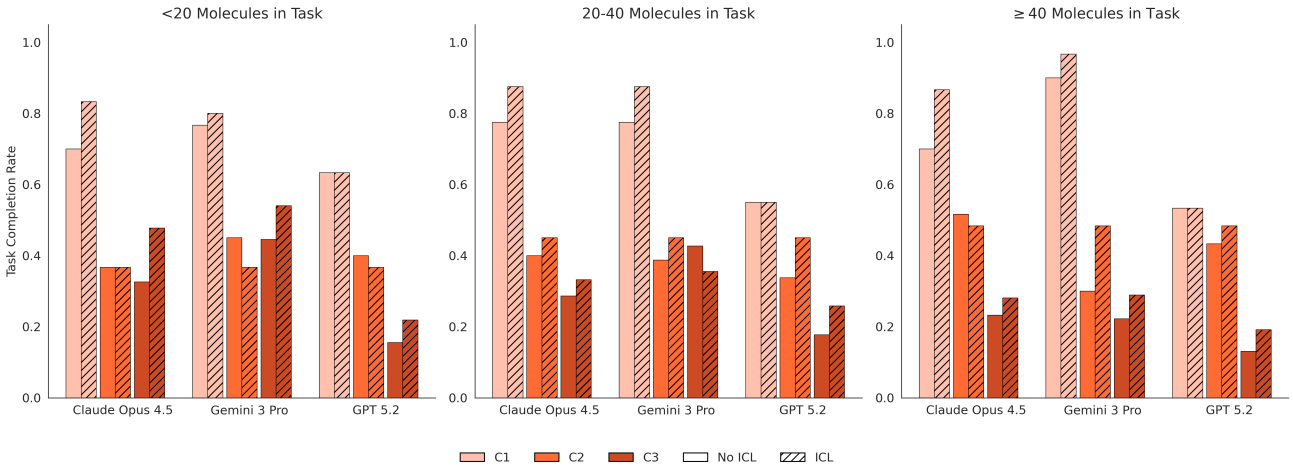


*Figure A.3.* Models are evaluated with one-shot in-context learning on 10% of questions from **REL**-C1,C2,C3.

### C.2. Test-Time Compute

In Table A.5, we investigate whether additional test-time compute can improve performance on difficult REL-A tasks.

*Table A.5.* Accuracy by number of thinking tokens on small examples of REL-A4 and REL-A5. Gains are small, but consistent.

| **Digits** | **GPT REL-A5** ($3 \times 3 \times 3$) | **Claude REL-A5** ($3 \times 3 \times 3$) | **GPT REL-A4** ($9 \times 9$) | **Calude REL-A4** ($9 \times 9$) |
|---|---|---|---|---|
| 4096 Tokens | 0.648 | 0.616 | 0.152 | 0.152 |
| 8192 Tokens | 0.664 | 0.624 | 0.168 | 0.144 |
| 16384 Tokens | 0.680 | 0.648 | 0.176 | 0.168 |

In Fig A.4 we show how test-time compute changes the task completion rate across **REL**-C1,C2,C3.

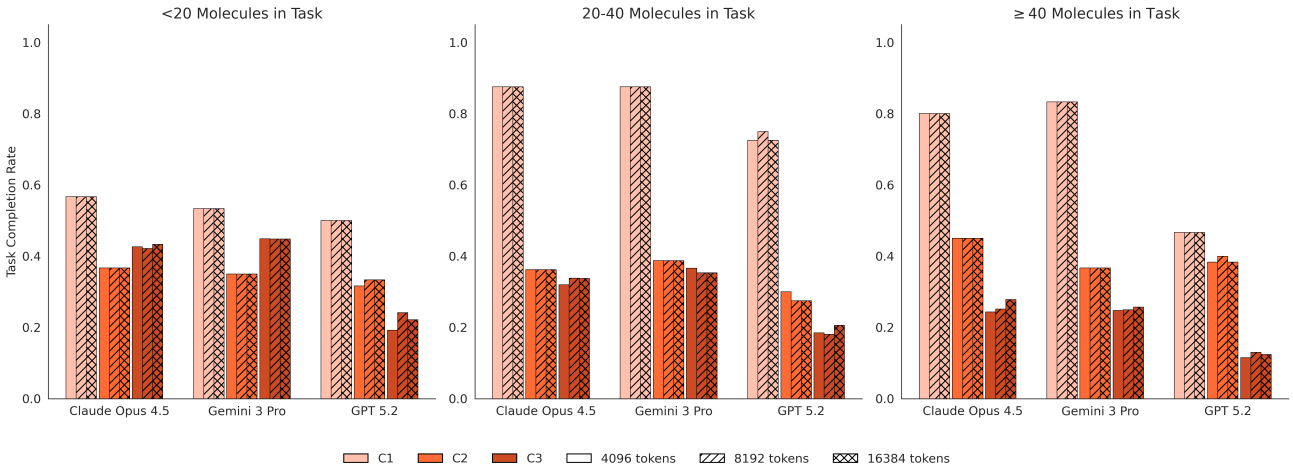


*Figure A.4.* Models are evaluated with different levels of test-time compute on 10% of questions from **REL**-C1,C2,C3.

### C.3. Operand Complexity

**Operand Complexity.** We vary the OC for REL-A3 and REL-A4 tasks by increasing the number of digits per entry in the RPM from 5 to 10 and then 20. As Table A.6 shows, moving from 5 to 10 digits results in only a small drop between 1 and 6% accuracy. Increasing this to 20 digits leads to catastrophic accuracy drops of as much as 50%, which seems to hint at the well-known difficulty that LLMs have when representing numbers (Zhou et al., 2025).

*Table A.6.* Accuracy by number of digits per entry on small examples of REL-A3 and REL-A4. Only for extremely large input values does accuracy plummet.

| **Digits** | **GPT REL-A3** ($9 \times 9$) | **Claude REL-A3** ($9 \times 9$) | **GPT REL-A4** ($3 \times 3$) | **Calude REL-A4** ($3 \times 3$) |
|---|---|---|---|---|
| 5 Digits | 0.912 | 0.752 | 0.248 | 0.208 |
| 10 Digits | 0.872 | 0.688 | 0.184 | 0.192 |
| 20 Digits | 0.360 | 0.336 | 0.136 | 0.112 |

We perform an analogous manipulation of OC for tasks **REL**-B by varying the motif ratio. As shown in Table A.7, decreasing the motif ratio from 10–12.5% to 0–5% causes average model accuracy to collapse by 92%, approaching zero across all models. Accuracy is highest at intermediate motif ratios and degrades sharply when the motif occupies only a small fraction of the sequence, indicating that models struggle to identify relevant structure when most of the input is irrelevant. This mirrors known difficulties of LLMs on needle-in-a-haystack settings (Bianchi et al., 2025). Together, these results demonstrate that OC, independent of RC, exerts a strong and systematic influence on model accuracy.

*Table A.7.* Accuracy by motif ratio (motif size / sequence length). Performance collapses as operand complexity increases in small motif ratios.

| Motif Ratio | Gemini | GPT | Claude |
|---|---|---|---|
| 0–5% | 2.6% | 0.8% | 0.0% |
| 5–10% | 9.2% | 10.2% | 1.8% |
| 10–12.5% | 13.5% | 18.6% | 2.1% |
| 12.5–15% | 22.6% | 29.0% | 12.9% |
| 15–17.5% | 19.6% | 17.6% | 17.6% |
| 17.5–20% | 34.2% | 23.7% | 28.9% |
| 20–22.5% | 18.2% | 21.9% | 10.9% |
| 22.5–25% | 28.6% | 20.0% | 22.9% |
| 25–30% | 31.2% | 28.0% | 16.1% |

## D. Example Prompts

Below we provide example prompts of each of the tasks in **REL**.

### D.1. REL-Algebraic

**REL**-A1

```
Only return the missing number!
row 1:  633, 633, 633; row 2:  354, 354, 354; row 3:  761, 761,
Answer set:
Answer #0:  769
Answer #1:  781
Answer #2:  789
Answer #3:  761
Answer #4:  780
Answer #5:  752
Answer #6:  712
Answer #7:  743
```

**REL**-A4

```
Only return the missing number!
row 1:  723, 38, 761; row 2:  152, 204, 356; row 3:  233, 279,
Answer set:
Answer #0:  476
Answer #1:  502
Answer #2:  334
Answer #3:  255
Answer #4:  512
Answer #5:  417
Answer #6:  687
Answer #7:  780
```

**REL**-A5

```
Complete the Raven's progressive tensor:
Only return the missing number!
Slice l=0:
row 1:  3, 6, 5
row 2:  4, 8, 9
row 3:  1, 7, 9
Slice l=1:
row 1:  13, 24, 29
row 2:  17, 25, 31
row 3:  17, 22, ?
Slice l=2:
row 1:  76, 84, 114
row 2:  78, 88, 115
row 3:  77, 88, 112
Answer set:
Answer #0:  45
Answer #1:  7
Answer #2:  52
Answer #3:  32
Answer #4:  24
Answer #5:  30
Answer #6:  16
Answer #7:  63
```

**REL**-A6

```
Complete the Raven's progressive tensor:
Only return the missing number!
Slice l=0:
row 1:  3, 6, 5
row 2:  4, 8, 9
row 3:  1, 7, 9
Slice l=1:
row 1:  8, 7, 10
row 2:  3, 1, 2
row 3:  2, 9, ?
Slice l=2:
row 1:  8, 2, 3
row 2:  5, 0, 3
row 3:  9, 6, 10
```

```
Answer set:
Answer #0:  0
Answer #1:  1
Answer #2:  8
Answer #3:  3
Answer #4:  6
Answer #5:  5
Answer #6:  10
Answer #7:  9
```

### D.2. REL-Biology

**REL**-B1

```
Homoplasy refers to structured convergence:
pairs or groups of distantly related taxa that repeatedly share the same
nucleotide motifs
across many independent alignment columns more often than expected, while
other taxa
with similar overall sequences do not share those nucleotide motifs as
consistently.
Your job is to examine the entire alignment and provided tree and decide
whether such structured
homoplasy is likely to be present and which taxa are involved.
Alignment (FASTA; positions indexed from 1):
>6
GAGATAATCATTCGGGAGTCAATTCCAAAATCCGTTCGGGATGAATTGTCTATCTGCCCCGCTTCGTGAGTACCGCTAACTCCTCG
...  (rest of sequences) ...
Tree (Newick):
(((6:0.9078,(3:0.8576,46:0.6305):0.5442):0.4086,(((((12:0.6359,(5:0.3115,16:0.3136):
...  (rest of tree) ...
Return your answer as:  Yes/No and if Yes, list the taxa involved.
```

### D.3. REL-Chemistry

**REL**-C1

```
Is this list of molecules a set of constitutional isomers (same molecular
formula, different connectivity)?
SMILES:
1.  ClC1C(Cl)C1Cl
2.  CC(Cl)=C(Cl)Cl
3.  C=CC(Cl)(Cl)Cl
4.  ClC=CC(Cl)Cl
5.  ClCC=C(Cl)Cl
Return exactly one of:
<Yes>
```

```
or
<No>
No explanation.
```

**REL**-C2

```
Given the following list of SMILES, what is the largest *connected* common
chemical motif (maximum common substructure) present in every molecule?
Rules:
- The motif must be a single connected fragment.
- Do NOT tautomerize molecules.
- Ignore stereochemistry unless it is explicitly encoded and required.
SMILES:
1.  COc1ccc2c(c1)N(CC(C)CN(C)C)c1ccccc1S2
2.  CC(CN(C)C)CN1c2ccccc2Sc2ccccc21
3.  CCc1ccc2c(c1)N(CC(C)CN(C)C)c1ccccc1S2
4.  CSc1ccc2c(c1)N(CC(C)CN(C)C)c1ccccc1S2
5.  CC(CN(C)C)CN1c2ccccc2Sc2ccc(C#N)cc21
Return your final answer as a single SMILES wrapped exactly like:
<smiles>YOUR_SMILES_HERE</smiles>
No explanation.
```

**REL**-C3

```
Given the following list of constitutional isomers, complete the set by
identifying the missing constitutional isomers.
Given SMILES:
1.  FCCC1CC1
2.  C=C(F)C(C)C
3.  CCC1CC1F
4.  C=CCCCF
5.  C=CCC(C)F

Return the missing molecules as SMILES, one per line, each wrapped exactly
like:
<smiles>YOUR_SMILES_HERE</smiles>
No explanation.
```