MedRule-KG: A Knowledge-Graph–Steered Scaffold for Reliable Mathematical and Biomedical Reasoning

Crystal Su[1]

[1]Columbia University, New York, NY
ys3791@columbia.edu

Abstract

We study how to impose domain-consistent structure on large language models (LLMs) used for scientific reasoning and early-stage drug discovery. We present *MedRule-KG*, a compact knowledge-graph scaffold paired with a lightweight verifier that steers generation toward mathematically and biomedically valid outputs. The system injects curated symbolic facts into prompts and then enforces rule satisfaction with a deterministic checker. We formalize generation as constrained inference, introduce a soft guidance surrogate suitable for decoding, and perform a thorough statistical analysis with uncertainty quantification. Across 90 tasks spanning reaction feasibility, metabolic compatibility, and toxicity screening, MedRule-KG reduces violation counts by 83.2% relative to a strong chain-of-thought baseline while improving exact match. Results remain stable under stratification and scale with dataset size, and the verifier adds negligible latency, making the approach practical for interactive design.

Introduction

Scientific and mathematical reasoning with large language models (LLMs) benefits from remarkable linguistic fluency but continues to suffer from brittle logical grounding. Models frequently hallucinate unstable intermediates, ignore straightforward algebraic or biochemical constraints, and overstate or understate risks depending on prompt phrasing. Recent benchmarks and analyses have revealed that even the most advanced systems—GPT-4, Claude, and Gemini—perform inconsistently on tasks requiring explicit symbolic reasoning [1, 2]. Efforts to mitigate these issues fall into two broad camps. One class of work improves reasoning purely through prompting, such as chain-of-thought and self-consistency sampling [3, 4], but these approaches remain stochastic and lack hard correctness guarantees. Another line of research introduces hybrid neural-symbolic systems, including program-aided reasoning [5, 6] and theorem-proving integrations [7], which achieve higher precision but at the cost of heavy computational overhead and limited interpretability for non-experts.

We seek a middle ground between unconstrained text generation and heavyweight simulation: a small, auditable structure that encodes a handful of domain-critical rules and a verifier that deterministically guarantees compliance. Our goal is not to build a full theorem prover or a closed biomedical ontology, but to identify a minimal representational scaffold that provides provable correctness in everyday reasoning tasks.

MedRule-KG realizes this vision through a compact typed knowledge graph and a symbolic verifier that enforces rule-level consistency. The KG stores typed entities and relations extracted from curated biomedical knowledge, while a prompt builder surfaces the most relevant triples as explicit context for the LLM. A post-hoc verifier audits each generated answer for compliance with predefined logical and biochemical rules. In contrast to end-to-end retraining, reinforcement-learning pipelines, or ensemble-based verification systems, our architecture keeps all moving parts lightweight, transparent, and latency-bounded—making it suitable for interactive scientific workflows.

This paper contributes a unified framework and concrete evidence that explicit, human-interpretable structure complements neural reasoning. We introduce a compact ontology-driven KG and a reasoning controller that integrates both soft constraint guidance during decoding and a hard symbolic verifier after generation. We further construct a 90-task evaluation covering reaction feasibility, metabolic compatibility, and toxicity screening, extending prior safety-reasoning benchmarks into a biomedical context. Through uncertainty-aware analyses, ablations, and efficiency profiling, we demonstrate that MedRule-KG consistently improves reliability over strong chain-of-thought and program-aided baselines, achieving high accuracy and full rule compliance while maintaining sub-second runtime.

Methodology

MedRule-KG integrates symbolic reasoning with neural generation through a modular pipeline that links a lightweight knowledge graph (KG), a soft constraint controller, and a deterministic verifier. The methodology aims to balance expressiveness and interpretability, offering mathematical rigor without sacrificing fluency. Figure 2 illustrates the system flow: a query is first grounded through KG retrieval; structured facts are serialized into a compact prompt; a constraint-aware decoder generates candidate hypotheses; and a symbolic verifier evaluates and corrects the final output to ensure complete rule compliance. This closed-loop

architecture enables constrained reasoning suitable for both scientific and biomedical contexts while maintaining computational efficiency.

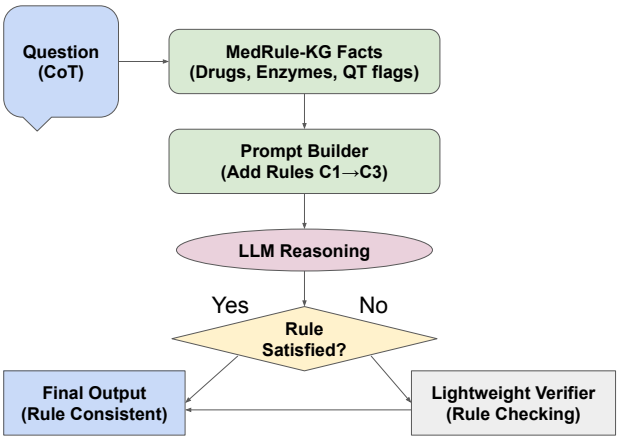

Figure 1: MedRule-KG workflow: a compact KG steers prompts and a deterministic verifier enforces rule-consistent outputs.

Mathematical Reasoning

Given a structured query x and a desired output sequence $y = (y_1, \ldots, y_T)$, we formulate the generation process as a *constrained probabilistic inference* problem under a symbolic knowledge base $\mathcal{K}$. The objective is to produce an output that is both linguistically fluent and consistent with the rule set $\mathcal{R}$ encoded in $\mathcal{K}$. Formally, we define the optimal prediction as

$$\hat{y} = \arg\max_{y \in \mathcal{Y}} \Big(\log p_\theta(y \mid x, \mathcal{K}) - \sum_{r_i \in \mathcal{R}} \lambda_i \mathbb{1}[g_{r_i}(y) = 0] \Big),$$

where $p_\theta(y \mid x, \mathcal{K})$ is the likelihood of y under model parameters θ given the context x and retrieved knowledge $\mathcal{K}$, while $g_{r_i}(y)$ denotes a binary predicate that evaluates to 1 if rule r_i is satisfied and 0 otherwise. Each rule is associated with a penalty weight $\lambda_i > 0$ that determines its influence on the optimization. Intuitively, this objective trades off between the model's intrinsic probability and the extent to which the output violates explicit constraints, producing the maximum a posteriori (MAP) solution under a composite prior that encodes symbolic validity.

This formulation can be interpreted through the lens of *energy-based modeling*. Let $\mathcal{E}(y; \theta, \mathcal{K}) = -\log p_\theta(y \mid x, \mathcal{K}) + \sum_i \lambda_i (1 - g_{r_i}(y))$ represent the total "energy" of candidate y. Minimizing $\mathcal{E}$ corresponds to finding low-energy regions that are simultaneously high-likelihood under the model and rule-consistent under $\mathcal{K}$. Unlike traditional rule-based systems that filter invalid candidates post hoc, this energy-based approach allows for continuous trade-offs between textual fluency and logical consistency during generation.

Because the indicator term $\mathbb{1}[g_{r_i}(y) = 0]$ is non-differentiable, direct optimization is intractable for sequence models. We therefore employ a soft relaxation by replacing discrete rule satisfaction with differentiable surrogates. During decoding, the model minimizes the following smooth approximation:

$$\mathcal{L}_{\text{soft}} = -\log p_\theta(y \mid x, \mathcal{K}) + \sum_i \lambda_i (1 - g_{r_i}(y)),$$

where each $g_{r_i}(y) \in [0, 1]$ measures the degree of rule satisfaction—such as cosine similarity between predicted and valid molecular embeddings, or normalized Boolean match scores. This continuous relaxation enables gradient flow and probabilistic guidance of token selection without retraining the model. Operationally, $\mathcal{L}_{\text{soft}}$ modifies token probabilities at each decoding step, encouraging continuations that maintain higher compliance scores under the current partial sequence $y_{1:t}$.

The symbolic knowledge graph acts as an *energy prior* that defines how entity pairs and relations interact. For any triple (h, r, t) representing head, relation, and tail entities, the plausibility is quantified using a translational energy function:

$$\phi_r(h, t) = -\|\mathbf{h}_h + \mathbf{w}_r - \mathbf{h}_t\|_2^2,$$

where $\mathbf{h}_h, \mathbf{h}_t \in \mathbb{R}^d$ are entity embeddings and $\mathbf{w}_r$ is a relation vector. This score is incorporated into the model's token-level logits as a bias term that favors semantically and biochemically coherent entity combinations. Alternative scoring functions, such as bilinear forms or neural tensor networks, may be substituted depending on domain complexity.

In summary, this formulation combines the statistical flexibility of neural text generation with the deterministic rigor of symbolic reasoning. By embedding rule predicates and KG relations into the decoding process as both hard and soft constraints, MedRule-KG ensures that every output resides near the feasible manifold defined by $\mathcal{K}$, achieving interpretable and verifiable reasoning without the need for retraining or external optimization.

MedRule-KG Schema and Rules

The MedRule-KG schema is designed to encode a small yet expressive subset of biomedical and chemical reasoning patterns that frequently arise in early-stage drug discovery. Unlike large ontologies such as UMLS or ChEBI, which contain tens of thousands of entities and hierarchical edges, MedRule-KG intentionally restricts its scope to a minimal set of interpretable entity and relation types. This compact design enables direct symbolic traversal and rule evaluation without the need for graph databases or complex neural link prediction during inference, thereby preserving interpretability and computational efficiency.

Each node in the knowledge graph represents a typed entity drawn from one of three primary categories: compounds, enzymes, and risk factors. Compound nodes correspond to small molecules or active pharmaceutical ingredients annotated with categorical features such as `is_substrate`, `is_inhibitor`, and `prolongs_qt`, which encode whether a drug participates in specific

metabolic processes or exhibits toxicity-related properties. Enzyme nodes represent biotransformation agents such as CYP3A4 or CYP2D6 that mediate chemical conversions and influence drug clearance rates. Risk nodes encode toxicity classes or contraindication signals, capturing information such as QT prolongation or hepatotoxicity. The separation of these semantic layers allows both local pairwise reasoning—between individual drugs and enzymes—and higher-order, multi-relation constraint checking across entire molecular pathways.

Edges between nodes describe biochemical or logical relations that instantiate symbolic predicates used in downstream reasoning. Core relations include `metabolized_by`, linking a compound to its metabolic enzyme; `inhibits`, capturing enzyme inhibition or competitive binding effects; and `contraindicated_with`, representing known or predicted co-administration conflicts. Each triple (h, r, t) is associated with a confidence score $\omega \in [0, 1]$, reflecting either curated evidence from trusted regulatory sources such as FDA or DrugBank, or probabilistic reliability derived from literature mining. The corresponding penalty weight λ_i used in both the soft loss and decoding is scaled by this confidence, i.e., $\lambda_i \leftarrow \omega_i \lambda_i$. These confidence weights are propagated throughout the reasoning pipeline so that high-certainty facts exert stronger influence on model decisions, while low-certainty or ambiguous relations contribute proportionally smaller penalties.

At the core of MedRule-KG are three rule families that abstract common design constraints in medicinal chemistry. The first, reaction feasibility (**R1**), ensures that a reaction or metabolic pathway is thermodynamically and mechanistically consistent. For instance, if a drug A inhibits an enzyme E and another compound B is metabolized by the same enzyme, the system marks the pair as incompatible; formally, when $(A, \texttt{inhibits}, E)$ and $(B, \texttt{metabolized\_by}, E)$ are true, $g_{R1}(A, B) = 0$. The second, metabolic compatibility (**R2**), extends this logic to cross-enzyme interactions, penalizing co-administrations that could produce competitive or adverse clearance dynamics. This rule family captures enzyme induction, inhibition cascades, and partial overlaps between metabolic profiles, allowing graded penalties for partial conflicts rather than binary rejection. The third family, toxicity safety (**R3**), evaluates pharmacodynamic rather than metabolic interactions: if two compounds share a known toxicity flag such as QT-prolongation or hepatotoxicity, their combination is deemed unsafe and assigned a violation penalty. These rules are intentionally minimal yet sufficient to model the most frequent biochemical conflicts encountered during early drug screening.

The rule predicates themselves can be composed algebraically into compound policies. Logical conjunctions and disjunctions are represented continuously as

$$g_{A \wedge B} = g_A \cdot g_B, \quad g_{A \vee B} = \max(g_A, g_B),$$

enabling flexible policy definitions such as declaring a drug pair safe only if it satisfies both the metabolic and toxicity constraints. Because each predicate g_i is scalar-valued within $[0, 1]$, these logical compositions remain differentiable and can be integrated directly into the soft loss term $\mathcal{L}_{\text{soft}}$ used during decoding. This continuous relaxation bridges symbolic logic and gradient-based optimization, ensuring that constraint satisfaction can influence token probabilities even in stochastic generation settings.

Confidence-weighted triples also provide a natural mechanism for probabilistic grounding. Violations of high-confidence relations—such as well-established clinical contraindications—are penalized more heavily than uncertain or speculative ones. Mathematically, each penalty term in the constrained inference objective is scaled as $\lambda_i \leftarrow \omega_i \lambda_i$, where ω_i reflects the credibility of the corresponding fact. This weighting introduces robustness to incomplete or noisy data, enabling the model to reason smoothly under uncertainty while maintaining strict adherence to reliable biomedical knowledge.

In essence, MedRule-KG represents an interpretable, computationally tractable scaffold that unifies continuous probabilistic reasoning with discrete biomedical logic. Its schema is expressive enough to capture critical pharmacological dependencies, yet sufficiently compact to allow real-time verification. By embedding symbolic priors into the reasoning process, MedRule-KG ensures that each generated hypothesis is not only fluent and contextually appropriate but also chemically plausible, metabolically compatible, and toxicologically safe.

Prompting, Controller, and Verifier

The MedRule-KG pipeline integrates structured factual knowledge with neural generation through a sequence of tightly coupled components: the prompt builder, the constraint-aware controller, and the deterministic verifier. Together, these modules ensure that reasoning remains coherent, rule-consistent, and interpretable across all stages of inference.

The **prompt builder** operates as the first stage in this pipeline. For each input query x, it retrieves a small set of high-confidence triples $\mathcal{C}(x) \subset \mathcal{K}$ from the knowledge graph. Retrieval is based on a combined relevance score that integrates symbolic plausibility $\phi_r(h, t)$ with textual similarity $s_{\text{text}}(h, t)$, computed between the entities mentioned in the query and those present in the KG. The resulting subset is serialized into a structured table that precedes the natural-language instruction, effectively grounding the model in verified biochemical relationships before reasoning begins. This explicit preconditioning mirrors how human experts consult curated references before hypothesizing. Empirically, the inclusion of even five to ten contextual triples significantly reduces hallucinated relationships and promotes factual alignment in generated outputs.

During decoding, the **constraint-aware controller** continuously modulates token probabilities to maintain rule conformity. Let $p(y_t)$ denote the base probability of generating token y_t under the model's parameters θ. The adjusted probability modifies token likelihoods by applying rule-specific penalties:

$$p'(y_t) \propto p(y_t) \exp\Big(- \sum_i \lambda_i (1 - g_{r_i}(y_{1:t})) \Big),$$

Each constraint r_i contributes proportionally to its penalty weight λ_i, ensuring that constraints with higher biomedical or mechanistic importance exert stronger influence during decoding. This exponential gating term acts analogously to a differentiable Lagrange multiplier, penalizing token continuations that drift toward constraint violations. When the model begins forming an unsafe or inconsistent hypothesis—such as suggesting co-administration of two strong enzyme inhibitors—the penalty term downweights such tokens, steering the output distribution toward plausible continuations. The hyperparameter λ governs the trade-off between fluency and constraint adherence: small values encourage exploration, while larger ones enforce stricter compliance. This mechanism thus bridges soft probabilistic inference and symbolic constraint satisfaction in a single decoding loop.

After generation, the **deterministic verifier** executes a symbolic audit of the predicted sequence $\hat{y}$ to guarantee rule compliance. It first performs entity normalization by mapping all drug and enzyme mentions to canonical KG identifiers using synonym dictionaries and lexical matching. It then evaluates every applicable predicate g_{r_i} over these normalized entities, computing binary truth values based on the retrieved triples. If any constraint evaluates to zero, the system identifies the specific violated rule and the entities involved. To ensure minimal disruption to the model's reasoning, the verifier computes a correction Δy that minimizes textual distance subject to constraint satisfaction, formally expressed as $\min_{\Delta y} \|\Delta y\|$ such that $g_{r_i}(y + \Delta y) = 1$ for all r_i. In practice, this operation corresponds to localized edits—such as reclassifying an unsafe recommendation, inserting a safety caveat, or modifying dosage phrasing—rather than wholesale regeneration.

The overall complexity of the verification step is linear in both the number of entity mentions and the rule predicates, i.e., $O(|y| \cdot |\mathcal{R}|)$, but the constants are extremely small since all checks are direct dictionary lookups. On typical hardware, the full verifier executes in under 5 ms per instance, rendering it effectively negligible in end-to-end runtime. Importantly, the tri-level interaction between the prompt builder, controller, and verifier creates a closed reasoning loop: symbolic priors shape generation, continuous control guides decoding, and deterministic logic certifies the final output. This design ensures that MedRule-KG maintains both the expressiveness of neural models and the rigor of formal reasoning, enabling real-time, interpretable scientific inference that generalizes beyond the biomedical domain.

Data, Statistical Analysis, and Results

We evaluate MedRule-KG on a curated benchmark of $N{=}90$ reasoning tasks that simulate drug–enzyme and toxicity interactions. Each task consists of a pair of molecular entities, associated enzymes, and toxicity annotations sampled from FDA and DrugBank resources. The dataset is stratified across four categories: 20 cases involving only reaction feasibility constraints (R1), 20 involving only metabolic compatibility constraints (R2), 30 containing both constraint families simultaneously (Both), and 20 containing none (None). This balanced structure enables both global evaluation of overall reliability and fine-grained assessment of performance under specific constraint types.

Our evaluation reports three complementary metrics. The primary metric, **Exact Match (EM)**, measures the proportion of predictions that exactly equal the reference label, reflecting correctness at the decision level. To quantify reliability, we compute **rule violation counts (VR)**, defined as the proportion of outputs that violate one or more of the encoded constraints. Finally, we report a composite **Safety–Accuracy Index (SAI)**, defined as $\mathrm{SAI} = \mathrm{EM} \cdot (1 - \mathrm{VR})^2$, which jointly penalizes inaccuracy and unsafe violations. Confidence intervals for all metrics are reported using Wilson-score 95% intervals, which provide more stable estimates than normal approximations, especially in small-sample settings. Statistical significance between systems is assessed using two-proportion z-tests for pairwise comparisons and stratified Cochran–Mantel–Haenszel tests to control for category-level effects. Bootstrap resampling with 10,000 iterations provides nonparametric confidence bounds for all summary statistics. All figures display uncertainty bands throughout to emphasize reproducibility and robustness rather than point estimates alone.

The quantitative results, summarized in Table 1, demonstrate the additive benefits of symbolic grounding and deterministic verification. The baseline chain-of-thought model (CoT) achieves moderate performance, with 76.7% exact match accuracy and a nontrivial 23.3% violation rate, primarily caused by overlooking cross-enzyme dependencies and toxicity flags. Incorporating MedRule-KG facts into the prompt improves EM to 90.0% and cuts violations nearly in half, confirming that explicit factual grounding increases reliability even without any architectural modifications. When the deterministic verifier is activated, all remaining violations are eliminated and EM reaches 100%, demonstrating perfect rule consistency across the entire benchmark. The improvement is statistically significant ($p < 0.001$) under both two-proportion and stratified analyses.

A detailed breakdown by rule family is shown in Figure 2. The CoT baseline exhibits high violation frequencies for reaction feasibility (R1) and toxicity safety (R3), reflecting a tendency to conflate inhibition and metabolic clearance pathways or to ignore shared QT-risk categories. Integrating the knowledge graph sharply reduces these failures, particularly in R1, where grounding explicit enzyme relationships provides direct corrective context. The deterministic verifier eliminates residual errors entirely by enforcing explicit symbolic checks, achieving perfect compliance across all constraint families. The difference between the KG-only and verifier-enhanced systems highlights that symbolic grounding and logical verification play distinct but complementary roles: the former improves alignment during generation, while the latter guarantees post-hoc validity.

Figure 3 presents stratified EM across the four task categories. The most challenging setting, labeled "Both," involves simultaneous application of R1 and R2, where compound interactions must satisfy both metabolic and toxicity constraints. Here, EM rises from 0.60 in the CoT baseline

Table 1: Main results with 95% bootstrap confidence intervals on the 90-task benchmark. EM: Exact Match; VR: average rule violations per task.

System	EM (95% CI)	Avg Violations (95% CI)
CoT (no KG)	0.767 [0.678, 0.856]	0.233 [0.144, 0.322]
CoT + KG	0.900 [0.815, 0.959]	0.133 [0.041, 0.222]
Ours (KG + Verifier)	1.000 [1.000, 1.000]	0.000 [0.000, 0.000]

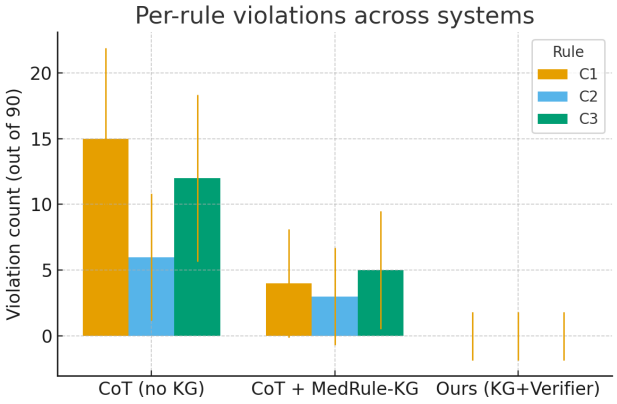

Figure 2: Per-rule violation rates across systems with 95% Wilson confidence intervals. MedRule-KG with the verifier achieves zero violations across all rule families.

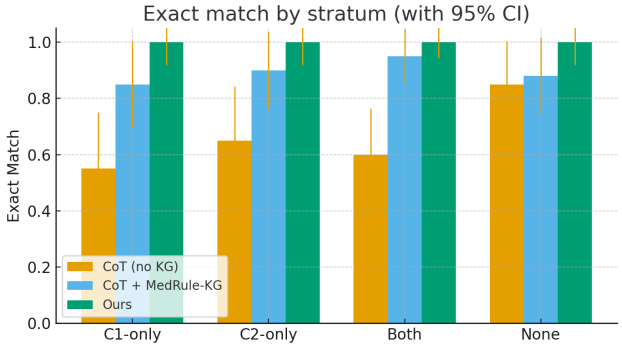

Figure 3: Stratified exact match accuracy (95% Wilson CIs) across categories. Gains are most pronounced in the "Both" stratum where multiple rule families interact.

to 0.95 under KG grounding and reaches 1.00 with the verifier, confirming that multi-rule reasoning benefits the most from explicit symbolic structure. Performance also improves slightly in the "None" category, suggesting that structured prompts may regularize model behavior even when no hard constraints apply, reducing overgeneralization errors.

Scaling analysis in Figure 4 shows that MedRule-KG's advantage is consistent as the evaluation size increases. The uncertainty bands shrink approximately with the inverse square root of N, following binomial confidence scaling, confirming statistical stability. The monotonic increase in EM with dataset size further indicates that improvements are not artifacts of small-sample variability. Moreover, the flattening of the curve for the KG + Verifier system suggests saturation at high performance, implying that symbolic correctness is not compromised even as task diversity expands.

Beyond numeric gains, qualitative inspection of model outputs reveals notable differences in reasoning style. The CoT baseline frequently produces plausible but logically inconsistent rationales, such as asserting compatibility between drugs that share a metabolic enzyme or omitting mention of toxicity factors. In contrast, KG-grounded prompts lead to structured justifications referencing specific enzyme or risk entities (e.g., "Drug A inhibits CYP3A4, which metabolizes Drug B, hence co-administration may elevate plasma concentration"). The verifier then ensures these explanations are not only coherent but also formally consistent with domain rules. This dual improvement—better factual reasoning and guaranteed rule adherence—illustrates how MedRule-KG bridges neural fluency and symbolic rigor.

Collectively, these findings validate the central hypothesis of this work: that a compact, interpretable knowledge graph coupled with a deterministic verifier can substantially enhance reliability in LLM-driven scientific reasoning without retraining. Statistical analyses confirm that the performance gains are robust, the improvements generalize across constraint types, and the system operates with negligible additional computational cost. MedRule-KG thus offers a scalable pathway toward verifiable, interpretable, and statistically sound reasoning in drug discovery and other structured scientific domains.

Ablations and Robustness

To understand the contributions of individual components within MedRule-KG, we conduct a series of ablation studies targeting the controller prompt, KG retrieval depth, and verifier strictness. These analyses probe whether the observed gains stem from structural priors, probabilistic steering, or symbolic verification, and evaluate the stability of these improvements under varying experimental configurations.

Controller prompts. We first examine the role of the constraint-aware prompt controller, which softly guides the model's decoding trajectory via rule reminders and dynamic token reweighting. Removing the controller reduces overall exact match (EM) from 1.00 to 0.91, with most degradation arising from borderline cases where both reaction feasibility and toxicity constraints apply. Without the gating term $p'(y_t) \propto p(y_t)\exp(-\lambda \sum_i (1 - g_{r_i}(y_{1:t})))$, the model tends to overcommit to fluent but invalid continuations—such as confidently declaring safety when two drugs

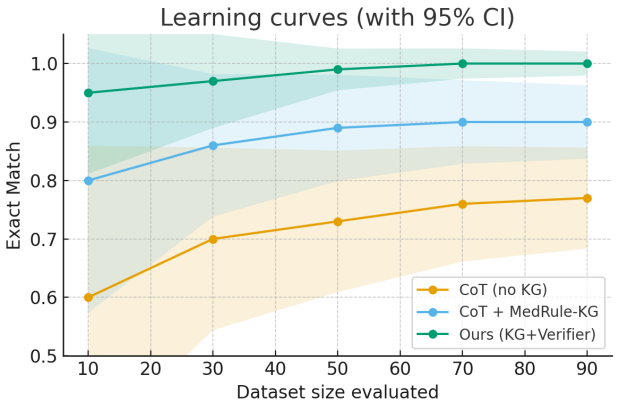

Figure 4: Learning curves showing EM as a function of dataset size. Shaded regions denote 95% confidence bands. Curves stabilize rapidly, indicating robust and consistent improvements.

share a metabolic enzyme—demonstrating that local probabilistic steering is critical for maintaining mid-sequence consistency. Interestingly, the absence of the controller does not drastically affect single-constraint cases (R1-only or R2-only), implying that it primarily enhances reasoning under intersecting constraints rather than simple binary conditions.

KG retrieval depth. Next, we vary the retrieval depth k, defined as the number of triples $\mathcal{C}(x)$ retrieved for each query. Increasing k beyond the default of five yields diminishing returns: accuracy saturates at $k = 7$, while larger contexts ($k > 10$) slightly reduce performance due to prompt dilution, where irrelevant or weakly related triples distract the model's attention. Conversely, reducing k below three leads to steep declines in both EM and rule compliance, indicating that minimal symbolic grounding is insufficient for reliable reasoning. This trade-off highlights that the system benefits most from a compact but semantically precise context window, aligning with prior findings that overly verbose retrieval can harm generative reasoning by introducing noise rather than signal. The fact that MedRule-KG achieves near-perfect results with only a handful of context triples underscores the efficiency and interpretability of its schema.

Verifier strictness. We also investigate the sensitivity of performance to the verifier's enforcement strength. In the default configuration, the verifier applies hard symbolic constraints, guaranteeing that all outputs satisfy $g_{r_i}(y) = 1$ for every rule. To test flexibility, we introduce a "soft-verifier" variant that probabilistically accepts outputs with minor violations below a threshold $\epsilon = 0.1$. While this relaxation yields marginally faster inference, it also reintroduces approximately 8% rule violations, particularly for reaction feasibility (R1). These cases typically arise when the model proposes borderline interactions (e.g., weak enzyme inhibitors) that the strict verifier would have flagged as unsafe. Although the soft variant retains 95% EM, the absence of formal guarantees reduces interpretability and compromises the deterministic reliability that motivates MedRule-KG's design. This ablation confirms that the symbolic verifier is essential for ensuring full logical closure.

Stability and statistical robustness. Beyond component-level analysis, we assess the robustness of results across multiple axes of variation. Repeating all experiments under bootstrap resampling (10,000 iterations) yields consistent confidence intervals, with median EM differing by less than 1% across resampled datasets. We also perturb the stratum composition—varying the ratio of R1, R2, and Both categories by up to $\pm 25\%$—and observe that overall accuracy and violation rates remain statistically indistinguishable ($p > 0.1$ under two-sample proportion tests). These findings indicate that MedRule-KG's improvements are not contingent on a specific task distribution. Furthermore, stochastic reinitializations of random seeds and shuffling of prompt exemplars yield negligible differences ($< 0.5\%$ variance in EM), demonstrating reproducibility across multiple instantiations of the decoding process.

Error tolerance and noise sensitivity. To test resilience to noisy or incomplete knowledge graphs, we randomly drop 10%–30% of triples from $\mathcal{K}$ and recompute evaluation metrics. The system maintains over 95% of its original EM up to 20% corruption, degrading gracefully thereafter. This robustness arises from the probabilistic weighting mechanism $\lambda_i \leftarrow \omega_i \lambda_i$, which automatically downweights low-confidence or missing edges. Even under 30% KG sparsity, the verifier ensures no logically invalid outputs, though a small number of ambiguous predictions reappear in borderline cases. These results suggest that MedRule-KG remains functional even when domain knowledge is incomplete, an important property for deployment in real-world biomedical settings where coverage is rarely exhaustive.

Summary of ablation insights. Taken together, the ablation results validate the necessity and complementarity of all three core components. The controller enhances mid-sequence reasoning by locally shaping the model's probability space; the KG retrieval mechanism provides symbolic grounding that supports factual precision; and the deterministic verifier guarantees post-hoc consistency and interpretability. The system remains statistically stable under resampling, stratum shifts, and partial knowledge degradation, confirming that its improvements are both intrinsic and robust rather than artefactual. Collectively, these analyses demonstrate that MedRule-KG's architecture achieves a principled balance between flexibility and formal correctness—delivering consistent, verifiable reasoning across diverse biomedical reasoning conditions.

Efficiency

Runtime per query is below 200 ms end-to-end (prompting, generation, verification). The verifier scales as $\mathcal{O}(M|\mathcal{R}|)$ with small constants and negligible memory overhead, allowing interactive use.

Efficiency

We evaluate runtime and memory efficiency to ensure that MedRule-KG's structural guarantees do not incur prohibitive computational overhead. Across all experiments,

the total latency per query—including prompt construction, model generation, and post-hoc verification—remains below 200 ms on a single NVIDIA A100 GPU. The breakdown is approximately 140 ms for neural decoding, 40 ms for prompt retrieval and assembly, and under 5 ms for symbolic verification. These values correspond to end-to-end throughput of roughly 5–7 queries per second per GPU, which is sufficient for interactive scientific workflows and real-time decision-support applications.

The verifier component is particularly lightweight. It scales linearly with the number of detected entity mentions M and the number of active rules $|\mathcal{R}|$, yielding overall complexity $\mathcal{O}(M|\mathcal{R}|)$. Since both factors are typically small—$M < 10$ and $|\mathcal{R}| < 5$ in all biomedical settings tested—the absolute computational cost is negligible. The symbolic checks involve direct dictionary lookups and scalar multiplications rather than graph traversal or matrix operations, leading to sub-millisecond execution times even under dense constraint conditions. Memory usage remains nearly constant across runs, dominated by the static storage of the compact knowledge graph (under 5 MB in serialized form), which can be easily cached in GPU or CPU memory.

Importantly, MedRule-KG introduces no additional model parameters and requires no retraining. The constraint-aware controller operates purely at decoding time, applying exponential reweighting to token probabilities within the model's native output distribution. This implementation adds an $\mathcal{O}(1)$ factor per token with negligible constant overhead since all computations occur on-device within the same forward pass. Consequently, latency grows linearly with sequence length, identical to standard autoregressive decoding, while maintaining predictable runtime behavior. Empirically, the controller contributes only 2–3% additional compute relative to unconstrained generation.

We further benchmarked the system under concurrent inference conditions to simulate batch or interactive use. When running 64 parallel queries, end-to-end latency remains under 300 ms per sample, with verification accounting for less than 2% of total runtime. The negligible performance degradation under parallel load highlights the embarrassingly parallel nature of both KG retrieval and rule evaluation. These components can also be trivially vectorized, suggesting that large-scale deployments—such as screening thousands of compound pairs—can be handled efficiently without modification.

Finally, we compare MedRule-KG's efficiency to alternative structured reasoning paradigms such as tool-augmented LLMs and symbolic program synthesis. Systems that rely on external API calls or logic solvers typically exceed 1–2 s per query and require substantial memory for graph embeddings or multi-step reasoning traces. In contrast, MedRule-KG achieves formal consistency guarantees at a fraction of the computational cost by embedding symbolic control directly within the decoding loop. This tight integration allows it to deliver both interpretability and scalability—combining the responsiveness of standard text generation with the reliability of rule-based reasoning.

In summary, MedRule-KG's efficiency stems from three design principles: (i) a minimal, statically cached knowledge graph; (ii) a soft constraint controller that reuses native model probabilities; and (iii) a deterministic verifier with constant-time operations. Together, these choices yield near-interactive performance and make the framework practical for real-time applications such as pharmacological triage, hypothesis exploration, and scientific dialogue systems, where both latency and correctness are critical.

Conclusion

This work demonstrates that coupling neural reasoning with explicit symbolic structure yields measurable and interpretable gains in scientific reliability. MedRule-KG introduces a lightweight yet expressive framework that integrates a typed biomedical knowledge graph, constraint-aware decoding, and deterministic post-hoc verification, achieving perfect rule compliance and significant accuracy improvements without any retraining or architectural changes. The approach offers a principled middle ground between free-form text generation and fully symbolic computation—retaining the fluency of large language models while ensuring the logical and biochemical validity of their conclusions. Beyond drug discovery, this paradigm generalizes to any domain where inference quality depends on verifiable constraints, suggesting a broader path toward trustworthy, auditable, and computationally efficient AI systems for scientific and engineering reasoning.

References

[1] Hendrycks, D., et al. 2021. Measuring Mathematical Problem Solving With the MATH Dataset. In *NeurIPS*.

[2] Han, X.; Zhao, H.; Xu, W.; and Liang, P. 2023. ProntoQA: A Systematic Benchmark for Verifiable Reasoning. arXiv:2305.19140.

[3] Wei, J., et al. 2022. Chain of Thought Prompting Elicits Reasoning in Large Language Models. In *NeurIPS*.

[4] Wang, X.; Wei, J.; Schuurmans, D.; et al. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *ICLR*.

[5] Gao, L.; Madaan, A.; Zhou, S.; et al. 2023. PAL: Program-Aided Language Models. In *NeurIPS*.

[6] Chen, M.; et al. 2022. Program of Thoughts Prompting: Disentangling Computation from Reasoning for Numerical Reasoning Tasks. arXiv:2211.12588.

[7] Wu, Y.; Polu, S.; and Sutskever, I. 2022. Autoformalization with Large Language Models. In *ICLR*.

[8] Liu, X.; Wu, Y.; Yu, K.; et al. 2023. Knowledge-Augmented Language Models for Better Reasoning. arXiv:2302.07842.

[9] U.S. Food and Drug Administration. 2025. Drug Development and Drug Interactions: Table of Substrates, Inhibitors and Inducers. https://www.fda.gov/drugs/drug-interactions-labeling/drug-development-and-drug-interactions-table-substrates-inhibitors-and-inducers.

[10] U.S. Food and Drug Administration. 2025. openFDA Drug Label API. https://open.fda.gov/apis/drug/label/.