Compartmentalised Agentic Reasoning for Clinical NLI

Maël Jullien[1,3], Lei Xu[3,5], Marco Valentino[4], André Freitas[1,2,3]
[1]Department of Computer Science, University of Manchester, UK
[2] National Biomarker Centre, CRUK-MI, University of Manchester, UK
[3]Idiap Research Institute, Switzerland
[4] School of Computer Science, University of Sheffield, UK
[5] École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
[3]{firstname.surname}@idiap.ch

Abstract

Large language models can produce fluent judgments for clinical natural language inference, yet they frequently fail when the decision requires the correct inferential schema rather than surface matching. We introduce *CARENLI*, a compartmentalised agentic framework that routes each premise–statement pair to a reasoning family and then applies a specialised solver with explicit verification and targeted refinement. We evaluate on an expanded CTNLI benchmark of 200 instances spanning four reasoning families: CAUSAL ATTRIBUTION, COMPOSITIONAL GROUNDING, EPISTEMIC VERIFICATION, and RISK STATE ABSTRACTION. Across four contemporary backbone models, *CARENLI* improves mean accuracy from about 23% with direct prompting to about 57%, a gain of roughly 34 points, with the largest benefits on structurally demanding reasoning types. These results support compartmentalisation plus verification as a practical route to more reliable and auditable clinical inference.

1 Introduction

Despite impressive surface-level performance on natural language inference (NLI) benchmarks (Bowman et al., 2015; Williams et al., 2018; Wang et al., 2019), large language models (LLMs) exhibit systematic failures in clinical reasoning when inference requires adherence to structured, domain-specific constraints, (Gururangan et al., 2018; McCoy et al., 2019; Ribeiro et al., 2020; Marcus and Davis, 2020; Agrawal et al., 2025; Jullien et al., 2023b, 2024); in settings where tolerance for systematic error is effectively zero, such failures have been associated with unsafe medical guidance and documented patient harm (Draelos et al., 2025; Eichenberger et al., 2025; Reddy and Reddy, 2025; Omiye et al., 2023; Goh et al., 2024; Amodei et al., 2016).

This limitation is captured by the diagnostic CTNLI benchmark of Jullien et al. (2025), which targets four core clinical reasoning capabilities: Causal Attribution, Compositional Grounding, Epistemic Verification, and Risk State Abstraction. Evaluation of six frontier large language models reveals systematic and reproducible violations of fundamental inference principles, resulting in a collapse of reasoning trajectories. Under a zero-tolerance evaluation regime, models achieve an average accuracy of only 0.25%. Even in this constrained setting, such performance demonstrates a fundamental misalignment between current LLM reasoning mechanisms and the requirements of clinical inference, necessitating substantive corrective intervention rather than incremental optimization.

This paper tests the hypothesis that CTNLI failures are driven by schema collapse, and that enforcing schema-conditioned, compartmentalised decision procedures can mitigate this failure. We propose *Compartmentalised Agentic Reasoning for Clinical NLI (CARENLI)* (Fig. 1), a direct intervention that decomposes inference into auditable roles: a *Router* selects the reasoning family, a family-specific *Solver* executes the corresponding procedure and emits an explicit trace, a *Verifier* checks premise grounding and procedural compliance, and a *Refiner* applies minimal corrections when the trace is partially correct but invalid. By explicitly binding inference to the appropriate schema, *CARENLI* enforces validity criteria that reasoning-agnostic prompting leaves implicit.

Our key contributions are:

Phenomenon. We characterise *schema collapse* as a dominant reasoning failure mode in CTNLI, in which models reuse a generic inference pattern across heterogeneous clinical reasoning demands, relying on surface-level pattern matching and producing fluent but constraint-violating reasoning trajectories.

Findings of the Association for Computational Linguistics: ACL 2026, pages 11205–11233
July 2-7, 2026

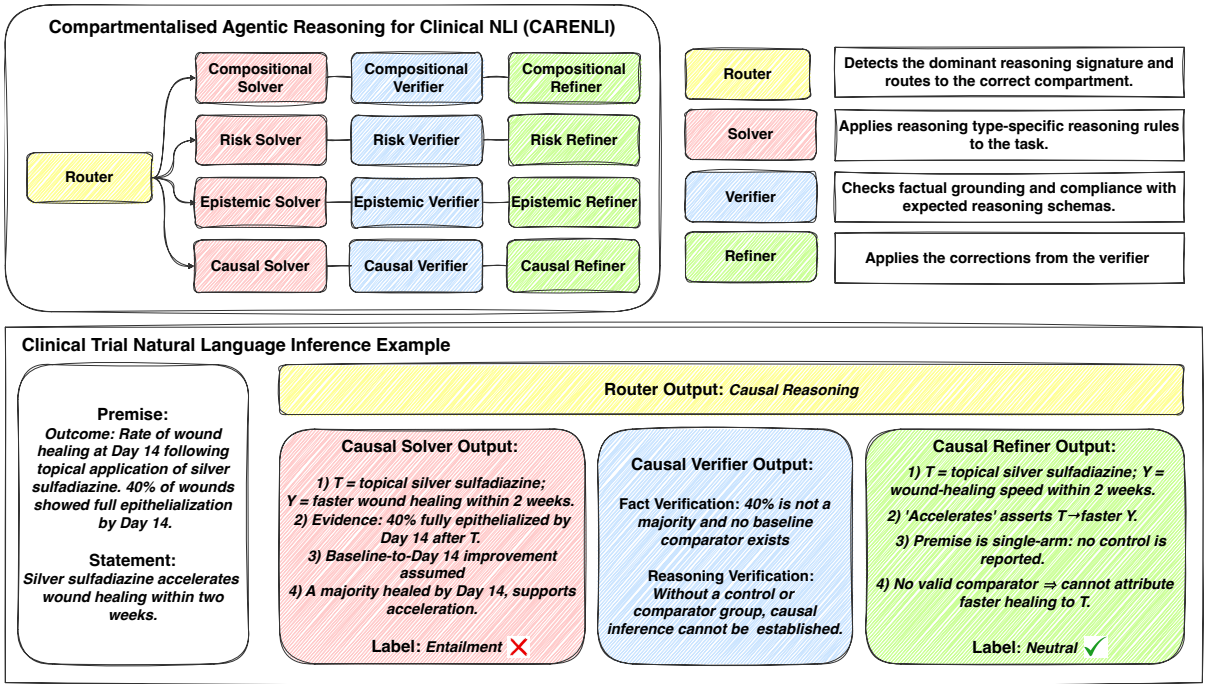

Figure 1: *CARENLI: Compartmentalised agentic reasoning framework for Clinical Trial NLI.* A *Router* assigns each premise–statement pair to a dominant reasoning type (Causal Attribution, Compositional Grounding, Epistemic Verification, or Risk State Abstraction). A reasoning-type-specific *solver* produces a provisional NLI label and explicit reasoning trace, which a *verifier* audits for factual grounding and schema compliance, and a *refiner* minimally corrects when needed. The figure illustrates the pipeline on a causal attribution example. Unlike generic prompting strategies *CARENLI* enforces formalised, structured reasoning trajectories that are explicitly grounded in clinical trial semantics and checked for logical consistency.

Mechanism. We introduce *CARENLI*, a compartmentalised agentic framework that performs reasoning-family routing, executes family-specific solver procedures with explicit traces, and verifies trace-level validity against the premise.

Validation. We extend and expert-validate a CTNLI benchmark, and demonstrate performance gains with *CARENLI* across four LLMs, with average improvements of approximately $+34$ percentage points in accuracy over generic prompting. We further analyse the contributions of *CARENLI*'s components and characterise cross–reasoning-type behaviour.

2 Methodology

2.1 Tasks and Dataset

This evaluation is grounded in the domain of *clinical natural language inference* (CTNLI), where the objective is to determine whether a candidate *statement* is entailed, contradicted, or left undetermined by a given *premise*. We adopt and extend the controlled CTNLI benchmark introduced by Jullien et al. (2025), which decomposes inference into four reasoning families, each defined by explicit inferential criteria:

i. CAUSAL ATTRIBUTION: distinguishes observational associations from causal claims, requiring assessment of temporality, comparators, and confounding.

ii. COMPOSITIONAL GROUNDING: evaluates whether clinical validity follows from the joint configuration of multiple interacting variables, such as dose, drug and schedule.

iii. EPISTEMIC VERIFICATION: tests whether a claim is supported by admissible evidence rather than authority, assertion, or unsupported diagnosis.

iv. RISK STATE ABSTRACTION: examines reasoning over latent or implicit clinical risks, particularly when severity and likelihood must be integrated.

Each reasoning type is formally specified through typed templates that constrain the inferential operations required for a correct decision. For

Type	N	Ent.	Contr.	Neut.
CAUSAL	50	0 (0%)	30 (60%)	20 (40%)
COMP.	50	0 (0%)	50 (100%)	0 (0%)
EPIS.	50	25 (50%)	25 (50%)	0 (0%)
RISK	50	15 (30%)	35 (70%)	0 (0%)
All	200	40 (20%)	140 (70%)	20 (10%)

Table 1: Label distribution by reasoning type.

instance, causal attribution items require explicit evaluation of comparator conditions, while compositional grounding items require compatibility checking across multiple clinical factors. All instances are instantiated from parametric templates, ensuring that variation is restricted to clinically meaningful dimensions, such as patient characteristics, or treatment regimens, while the underlying reasoning structure remains invariant.

The benchmark has been substantially extended from its original formulation, using the same data construction and validation protocol as Jullien et al. (2025). Table 1 reports the label distribution for each reasoning family.

The benchmark is not balanced across NLI labels within each family, as it is designed as a controlled diagnostic framework for isolating specific reasoning failures under tightly defined inferential regimes. As a consequence of these constructions, certain family–label combinations do not appear.

Details of the data generation pipeline, expert validation procedure, and inter-annotator agreement analysis are provided in Appendix A.3.

2.2 Compartmentalised Reasoning

Prior work on the CTNLI benchmark (Jullien et al., 2025) reveals a systematic dissociation between knowledge access and inference in large language models. Across all four reasoning families, models reliably encode the clinically relevant facts, as evidenced by near-ceiling performance on ground-knowledge probes, yet fail to apply these facts in accordance with the inferential criteria defining each task. The resulting errors are not stochastic. Instead, they are highly consistent within each reasoning type, indicating the repeated application of uniform heuristics rather than principled inference. Causal claims are reduced to temporal association, compositional validity to single-attribute matching, epistemic judgment to deference to assertion, and latent risk assessment to frequency-based reasoning. This pattern indicates a failure of inference deployment rather than knowledge acquisition.

These failures can be understood as a consequence of treating semantically heterogeneous inference problems as instances of a single undifferentiated reasoning process. The four CTNLI families correspond to distinct inferential schemas with non-interchangeable validity conditions. Causal attribution depends on counterfactual and intervention-based reasoning, compositional grounding on multi-factor compatibility constraints, epistemic verification on evidential hierarchies, and risk state abstraction on the integration of severity and likelihood. Both cognitive theories of reasoning and formal semantic analyses characterise these schemas as structurally distinct, each governed by its own representational requirements and decision rules (Sloman and Sloman, 2009; Johnson-Laird, 1983; Pearl, 2009). Collapsing these type-distinct relations into a single heuristic mode erases the constraints that make the inferences meaningful, yielding locally coherent but globally invalid judgments.

This diagnosis motivates the need for compartmentalised reasoning in clinical NLI. If errors arise because models apply a uniform inference strategy to tasks with incompatible structural demands, then improving fidelity requires enforcing explicit separation between reasoning families and constraining inference to the principles specific to each. Reasoning-agnostic prompting, including free-form Chain-of-Thought (CoT), provides no such separation and therefore permits heuristic shortcuts that systematically violate reasoning-type-specific criteria. By contrast, a compartmentalised design reinstates the boundaries between causal, compositional, epistemic, and risk-based inference, ensuring that each problem is solved using the appropriate inferential schema.

In the following section, we operationalise this principle through *Compartmentalised Agentic Reasoning for Clinical NLI (CARENLI)* (Fig 1). The framework enforces compartmentalisation by decomposing inference into specialised roles, enabling auditable, schema-aligned reasoning while preserving coordination across stages.

2.3 CARENLI

Compartmentalised Agentic Reasoning for Clinical NLI (*CARENLI*) instantiates compartmentalised reasoning through an agentic pipeline that enforces separation between problem recognition, inference execution, validation, and correction (Fig 1).

Formally, given a premise–statement pair (p, s),

CARENLI computes a final entailment judgment $\bar{y}$ via a sequence of specialised agents:

$$(p, s) \xrightarrow{\mathcal{R}} F \xrightarrow{\mathcal{S}_F} (y, \tau) \xrightarrow{\mathcal{V}_F} (v, c_f, c_p) \xrightarrow{\mathcal{R}_F} (\bar{y}, \bar{\tau}).$$

The router $\mathcal{R}$ assigns the instance to a reasoning type F. A type-specific solver $\mathcal{S}_F$ produces a provisional label y with an explicit reasoning trace τ. The verifier $\mathcal{V}_F$ audits this trace for factual grounding and schema compliance, emitting a violation signal v and structured critiques (c_f, c_p). When required, the refiner $\mathcal{R}_F$ applies these critiques through minimal edits to yield the final judgment $\bar{y}$ and trace $\bar{\tau}$ (illustrated in Fig. 1).

Each routed family is tied to a distinct *inference trajectory* that specifies what a valid intermediate trace must do. In EPISTEMIC VERIFICATION, the trace must adjudicate conflicting claims by privileging stronger evidence; in CAUSAL ATTRIBUTION, it must distinguish interventional contrast from observational association; in COMPOSITIONAL GROUNDING, it must test the admissibility of the full clinical configuration rather than isolated factors; and in RISK STATE ABSTRACTION, it must compare harms by jointly weighing likelihood and severity. We therefore anchor CARENLI's prompts in family-specific decision procedures rather than free-form explanatory text. The formal solver characterisations are provided in Appendix A.1, and the exact router, solver, verifier, and refiner prompts are reported in Appendix A.5 (Fig. 17; Figs. 5–16).

Router. $(p, s) \xrightarrow{\mathcal{R}} F$ The Router $\mathcal{R}$ implements compartmentalisation at the level of problem recognition. It maps each (p, s) to exactly one reasoning type $F \in \mathcal{F}$, where $\mathcal{F} = \{$CAUSAL ATTRIBUTION, COMPOSITIONAL GROUNDING, EPISTEMIC VERIFICATION, RISK STATE ABSTRACTION$\}$.

Solver. $(p, s) \xrightarrow{\mathcal{S}_F} (y, \tau)$ The solver $\mathcal{S}_F$ performs reasoning-type-specific inference. Conditioned on the selected reasoning type F, it applies the corresponding decision procedure and reasoning principles to produce a provisional entailment label y and an explicit reasoning trace τ. Each solver is constrained by the normative criteria of its reasoning type, such as comparator requirements in CAUSAL ATTRIBUTION, multi-factor compatibility in COMPOSITIONAL GROUNDING, evidential hierarchies in EPISTEMIC VERIFICATION, or severity–likelihood integration in RISK STATE ABSTRACTION. Full solver specifications are provided in Appendix A.1, with the corresponding prompt templates in Appendix A.5.

Verifier. $(y, \tau) \xrightarrow{\mathcal{V}_F} (v, c_f, c_p)$ Verification improves LLM reliability by surfacing unsupported claims and schema violations (Webson and Pavlick, 2022; Quan et al., 2024). Given (y, τ), $\mathcal{V}_F$ performs fact verification, identifying non–premise-grounded or clinically inadmissible claims (Gravel et al., 2023; Aljamaan et al., 2024), and pattern verification, checking compliance with the reasoning-type-specific inferential schema. It outputs a violation signal v and structured fact- and pattern-level critiques (c_f, c_p) for downstream correction.

Refiner. $(y, \tau, v, c_f, c_p) \xrightarrow{\mathcal{R}_F} (\bar{y}, \bar{\tau})$ Conditioned on the violation signal v, the refiner $\mathcal{R}_F$ applies the verifier's critiques (c_f, c_p) via minimal, schema-preserving edits to the solver's reasoning trace τ, producing a final trace $\bar{\tau}$ and judgment $\bar{y}$.

3 Empirical Evaluation

We evaluate the *CARENLI* framework across four contemporary large language models: GPT-5.1 (OpenAI, 2025), GPT-4.1 and GPT-4o-mini (Hurst et al., 2024), and DeepSeek R1 (Guo et al., 2025).

Three evaluation settings are considered, each designed to isolate a distinct source of performance gain.

First, the full *CARENLI* framework activates all components of the pipeline; routing over reasoning families, solver specialisation through family-specific agents, and post-hoc verification and refinement. This condition instantiates generic agentic mechanisms with specialised components and measures their combined effect in the CTNLI setting.

Second, an *Oracle* CARENLI ablation bypasses routing by supplying the correct reasoning family directly to the downstream solver. Although not intended as a deployable condition, it serves as a diagnostic upper bound on routing quality and isolates the contribution of specialised schema-conditioned reasoning under perfect family identification.

Third, baseline prompting conditions evaluate the same models under generic prompting alone, using either chain-of-thought prompting (**Agnostic CoT**) or direct answering (**Agnostic Direct**). These prompts are reasoning-family agnostic and provide no guidance toward the family-specific inferential schemas encoded in CARENLI. Together,

these three settings decompose gains due to generic multi-step prompting, specialised reasoning procedures, and routing accuracy within the full pipeline..

Each model–strategy pair is evaluated over the full dataset (Section 2.1) across 4 runs; the Oracle *CARENLI* ablation is run once.

4 Results

Figs 2–4 and Tables 2—5 show that *CARENLI* substantially improves CTNLI performance by enforcing reasoning-type-specific decision procedures rather than reasoning-agnostic prompting. Aggregated across models and task families, *CARENLI* reaches 56.8% macro accuracy, compared to 22.5% for agnostic CoT and 23.1% for agnostic Direct, an absolute gain of roughly +34%. Improvements are consistent across all four backbones, though heterogeneous in magnitude (e.g., GPT-5.1 rises from 24% to 44%), reinforcing that scale alone does not guarantee schema-faithful clinical reasoning.

Gains are largest for RISK STATE ABSTRACTION and EPISTEMIC VERIFICATION, where *CARENLI* attains 92.2% and 62.7% average accuracy and recovers from near-collapse baselines (Risk improves by +51.9% relative to the strongest agnostic condition). CAUSAL ATTRIBUTION improves more modestly to 46.5%, while COMPOSITIONAL GROUNDING remains the weakest regime despite improvement to 25.6%, indicating persistent compositional failures in some backbones. Component analyses attribute most of the gain to the schema-constrained solver (*Oracle Solver*: 60.5%), with routing as the main bottleneck (70.8% when correctly routed vs. 21.1% when misrouted). Verifier and refiner stages contribute very marginal, task-dependent gains (overall refinement +0.7%), primarily when solver trajectories are partially correct but procedurally incomplete.

Qualitatively, *CARENLI* shifts model behavior from plausibility narratives toward schema-aligned traces that explicitly instantiate the reasoning type's decision criteria, making typical agnostic failures traceable to specific missing checks (e.g., comparator enforcement, cross-factor constraints, or severity-weighted risk integration).

4.1 Consistency Across Backbone Models

Schema-constrained prompting consistently outperforms reasoning-agnostic baselines by an average of 34% across diverse backbones. As shown in Fig 2 and Table 5, *CARENLI* outperforms the stronger of the two agnostic prompting baselines for every evaluated model, yielding macro accuracy improvements of 37.3% for deepseek-r1, 36.8% for gpt-4.1, 19.4% for gpt-5.1, and 38.2% for gpt4o-mini. While the magnitude of improvement varies with model capacity, no backbone exhibits a regression relative to prompting baselines.

Model scale alone is an unreliable predictor of clinical reasoning accuracy. The smallest improvement is observed for gpt-5.1, from 24.5% to 43.9%. The limited improvement observed for gpt-5.1 is consistent with prior findings that increased model scale does not reliably translate into improved clinical reasoning performance (Jullien et al., 2025).

4.2 Performance by Reasoning Type

***CARENLI* amplifies reasoning competence where inferential structure is tractable, but leaves residual failure modes in structure-intensive regimes.** Fig 2 and Table 5 enable a reasoning-type–level analysis of *CARENLI*'s effectiveness. Averaged across models, *CARENLI* achieves 92.2% accuracy on RISK ABSTRACTION, 62.7% on EPISTEMIC VERIFICATION, 46.5% on CAUSAL ATTRIBUTION, and 25.6% on COMPOSITIONAL GROUNDING.

Relative to the strongest agnostic baseline (CoT or Direct prompting, averaged across models), these results correspond to absolute gains of +51.9% on RISK ABSTRACTION, +40% on EPISTEMIC VERIFICATION, +15% on CAUSAL ATTRIBUTION, and +20% on COMPOSITIONAL GROUNDING.

The observed gains do not track label imbalance. As shown in Table 1, if improvements were driven by majority-class bias toward contradiction, the largest gains would be expected in COMPOSITIONAL GROUNDING (100% contradiction), followed by RISK ABSTRACTION (70%), Causal (60%), and the smallest gains in EPISTEMIC VERIFICATION (50/50). Instead, gains are largest in RISK ABSTRACTION (+51.9%) and EPISTEMIC VERIFICATION (+40%), exceed those in COMPOSITIONAL GROUNDING (+20%), and do not follow the imbalance gradient. This pattern is inconsistent with a majority-label strategy and instead indicates improvements aligned with reasoning-type structure rather than label frequency.

These gains are uneven across reasoning regimes and models. In some settings, *CARENLI* recovers

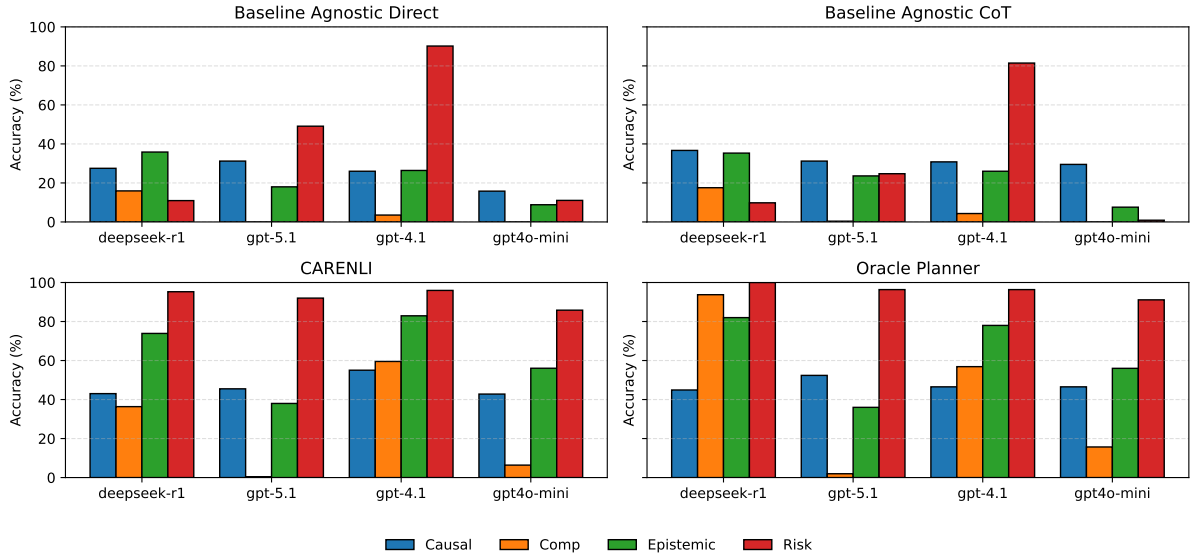

Figure 2: Overall accuracy on CTNLI tasks across all models and evaluation strategies (*CARENLI*, *Oracle Router*, *CoT*, and *Direct*). Results are averaged over four runs per configuration.

performance from near-baseline failure. For example, COMPOSITIONAL GROUNDING accuracy for GPT-4.1 increases from 3.5% under agnostic Direct prompting to 59.5% under *CARENLI*. In contrast, for gpt-5.1, *CARENLI* yields no improvement on the same reasoning type, with accuracy remaining at 0.40%.

When reasoning families are ranked by average agnostic baseline accuracy and by average *CARENLI* accuracy (Fig 2 and Table 5), the ordering is identical: RISK ABSTRACTION, EPISTEMIC VERIFICATION, CAUSAL ATTRIBUTION, COMPOSITIONAL GROUNDING. In addition, absolute *CARENLI* gains exhibit a positive correlation with baseline agnostic performance (Pearson's $r \approx 0.54$), indicating that reasoning regimes with higher initial accuracy under agnostic prompting tend to benefit more from *CARENLI*.

***CARENLI* amplifies existing latent competence rather than creating new reasoning capability.** These results indicate that *CARENLI* amplifies latent reasoning competence rather than transforming underlying reasoning capacity. Structured prompting yields substantial gains where partial inferential structure is already expressed under agnostic prompting, but does not invert task difficulty or induce robust performance in regimes characterized by near-zero baseline accuracy. *CARENLI* primarily improves execution and constraint adherence, rather than introducing new compositional reasoning abilities where they are largely absent.

4.3 Component-wise Analysis

4.3.1 Router

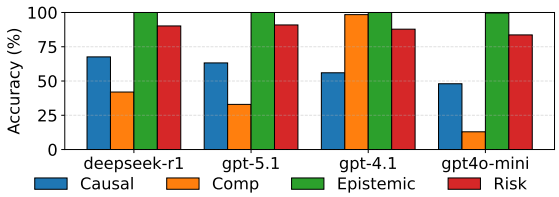

Figure 3: Overall accuracy of reasoning classification across models.

Incorrect reasoning-type routing induces downstream performance collapse. Fig 3 and Tables 2 and 6 summarise the Router's behaviour. Averaged across all four backbones and all four reasoning families, the Router achieves 73.3% classification accuracy, but performance is highly uneven across reasoning types: EPISTEMIC VERIFICATION is near-ceiling (99.6–100%) and RISK ABSTRACTION is consistently high (83.6–90.9%), whereas CAUSAL ATTRIBUTION is only moderate (48.0–67.6%) and COMPOSITIONAL GROUNDING is the least stable (12.9–98.4%). The confusion matrix further indicates that misrouting is systematic rather than random, with particular concentration of errors in compositional cases.

Routing quality has a large downstream effect. Averaged across all models and reasoning types, solver accuracy is 70.8% on correctly routed items

	Causal Attr.			Comp. G.			Epis. Verif.			Risk Abstr.		
Model	**Correct**	**Wrong**	**Δ**	**Correct**	**Wrong**	**Δ**	**Correct**	**Wrong**	**Δ**	**Correct**	**Wrong**	**Δ**
deepseek-r1	50.3 (68.4%)	27.3 (31.6%)	↑+23.0	86.0 (42.3%)	0.0 (57.7%)	↑+86.0	73.9 (100.0%)	– (0.0%)		99.2 (90.2%)	59.3 (9.8%)	↑+39.9
gpt-5.1	59.5 (62.7%)	22.0 (37.3%)	↑+37.5	1.2 (35.9%)	0.0 (64.1%)	↑+1.2	38.0 (100.0%)	– (0.0%)		97.2 (90.9%)	40.0 (9.1%)	↑+57.2
gpt-4.1	59.5 (55.0%)	49.5 (45.0%)	↑+10.0	60.2 (98.8%)	0.0 (1.2%)	↑+60.2	82.9 (100.0%)	– (0.0%)		100.0(87.8%)	66.7 (12.2%)	↑+33.3
gpt4o-mini	43.3 (48.0%)	42.3 (52.0%)	↑+1.0	45.5 (14.0%)	0.0 (86.0%)	↑+45.5	55.9 (99.6%)	100.0(0.4%)	↓-44.1	93.5 (83.6%)	46.7 (16.4%)	↑+46.8

Table 2: Accuracy by reasoning family, split by routing outcome (Wrong if predicted reasoning-type $\neq$ ground truth). Cells report solver accuracy (%), with routing prevalence shown as (% of items). Δ is Correct minus Wrong.

versus 21.1% on misrouted items, an absolute gap of 49.7%. This establishes the Router as a significant bottleneck: incorrect reasoning type selection typically collapses performance, especially for regimes that require strict schema adherence.

Correct routing is necessary, but not sufficient to achieve consistently correct inference. However, planning is not sufficient for success, as some failures persist even when routing is correct (e.g., gpt-5.1 achieves only 1.2% on correctly routed COMPOSITIONAL GROUNDING items), indicating additional downstream inference limitations.

4.3.2 Solver

Backbone models can effectively execute solver inference schemas that are aligned with the intended reasoning operations, providing the majority of CARENLI's gains over reasoning-agnostic baselines. We isolate solver effectiveness using the *Oracle Solver* setting (Table 3). Averaged across models and reasoning families, *Oracle Solver* attains a macro accuracy of 60.5%, substantially outperforming agnostic baselines (Agnostic CoT: 22.5%; Agnostic Direct: 23.1%; Fig 2 and Table 5). In comparison, *CARENLI* Solver achieves 56.1% macro accuracy. Therefore, the majority of the performance gain from *CARENLI* is attributable to the solver's structured inference schemas.

As shown in Section 4.3.1, solver accuracy is highly sensitive to correct reasoning type routing, with substantial degradation under misrouting (Table 2). This sensitivity suggests that the solver schemas encode meaningful structural constraints that closely correspond to the intended reasoning operations. This demonstrates that models are capable of following explicitly defined inference schemas, and that these schemas are well aligned with correct reasoning behavior.

Generic inferential scaffolding underlying the reasoning-type specific solver schemas improves reasoning even under misrouting. Notably, however, misrouted solvers remain non-trivially effective. In the *CARENLI* setting, incorrectly routed solvers achieve 34.9% average accuracy, exceeding both agnostic baselines (Tables 2, and 5). This suggests that generic procedural scaffolding, such as explicit decision steps, comparator checks, and grounding requirements, confers a measurable inference benefit even when reasoning-type-specific constraints are mismatched. Nevertheless, the substantial gap relative to correctly routed performance confirms that such generic structure is insufficient to substitute for the appropriate reasoning schema.

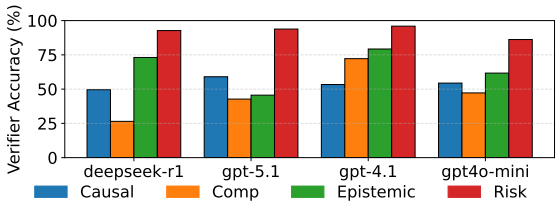

Figure 4: Verifier accuracy across reasoning families and models.

4.3.3 Verifier

Verification reliability tracks reasoning-type difficulty. Fig 4 and Table 7 quantify the effectiveness of the verifier across models and reasoning families. Averaged across all backbones and tasks, verifier accuracy under *CARENLI* is 62.1%, indicating that the verifier is moderately reliable and not merely echoing solver decisions but performing a distinct auditing function.

Verifier accuracy is highest for RISK STATE ABSTRACTION (macro average 92.4%) and EPISTEMIC VERIFICATION (64.9%), while being lower for CAUSAL ATTRIBUTION (54.1%) and COMPOSITIONAL GROUNDING (47.2%). This pattern mirrors solver difficulty across reasoning families and indicates that verifier errors concentrate in regimes where solver trajectories themselves are less well-formed.

	Causal Attr.			Comp. G.			Epis. Verif.			Risk Abstr.			Refinement Dyn.	
Model	**Sol.**	**Ref.**	**Δ**	**Sol.**	**Ref.**	**Δ**	**Sol.**	**Ref.**	**Δ**	**Sol.**	**Ref.**	**Δ**	**Trig.%**	**Flip.%**
CARENLI														
deepseek-r1	38.1	43.0	+4.9↑	40.3	36.4	-3.9↓	79.9	73.9	-6.0↓	96.4	95.3	-1.1↓	25.1	11.6
gpt-5.1	37.7	45.5	+7.8↑	1.7	0.4	-1.3↓	53.6	38.0	-15.6↓	92.4	92.0	-0.4↓	44.8	22.4
gpt-4.1	50.0	55.0	+5.0↑	48.4	59.5	+11.1↑	85.0	82.9	-2.1↓	95.9	95.9	0.0	17.4	14.3
gpt4o-mini	32.8	42.8	+10.0↑	4.3	6.4	+2.1↑	59.7	56.0	-3.7↓	81.5	85.8	+4.3↑	41.3	26.6
Oracle Router														
deepseek-r1	44.9	44.9	0.0	91.7	93.8	+2.1↑	80.0	82.0	+2.0↑	100.0	100.0	0.0	24.5	8.2
gpt-5.1	42.9	52.4	+9.5↑	3.9	2.0	-1.9↓	46.0	36.0	-10.0↓	96.4	96.4	0.0	43.9	19.7
gpt-4.1	44.2	46.5	+2.3↑	49.0	56.9	+7.9↑	82.0	78.0	-4.0↓	96.4	96.4	0.0	10.6	9.4
gpt4o-mini	32.6	46.5	+13.9↑	5.9	15.7	+9.8↑	62.0	56.0	-6.0↓	91.1	91.1	0.0	35.3	23.5

Table 3: Solver vs Refiner accuracy with deltas. Trig.% denotes instances where the verifier rejected the solver output. Flip.% denotes instances where the refined prediction differed from the solver prediction.

4.3.4 Refiner

Refinement offers marginal gains. Table 3 quantifies the impact of refinement on solver outputs. Averaged across all models and reasoning families, refinement yields a marginal absolute accuracy change of +0.7% under *CARENLI*, indicating that the refiner corrects a non-trivial subset of solver errors while leaving the majority of predictions unchanged.

Refinement effectiveness is highly reasoning-type dependent. The largest gains are observed for CAUSAL ATTRIBUTION, with an average improvement of +6.9%. In contrast, refinement has near-zero or negative effect for RISK STATE ABSTRACTION, where solver accuracy is already near ceiling, and for EPISTEMIC VERIFICATION, where over-correction occasionally reduces performance (e.g., gpt-5.1 drops from 53.6% to 38.0%).

Overall, the refiner is not a primary performance driver and does not constitute a bottleneck in the *CARENLI* pipeline. Its impact is bounded by the quality of the verifier signal and the structural correctness of the solver trajectory, particularly under misrouting or weak compositional reasoning. The concentration of refinement gains in regimes with moderate, rather than ceiling, verifier accuracy provides indirect validation that corrections are applied to meaningful, correctly identified violations rather than introduced arbitrarily.

4.4 Qualitative Analysis

Schema-constrained inference replaces plausibility narratives with explicit, checkable decision criteria, improving both end-task performance and intermediate reasoning trajectories.

Qualitative inspection of representative items reveals a pronounced divergence between *agnostic CoT* and *structured* (Oracle Router) inference, not merely in final labels but in the *reasoning trajectories* that produce them (see Section A.2 for a detailed analysis). Under Oracle Router routing, solver traces are consistently organised around the reasoning-type-specific decision criteria of the CTNLI framework. In contrast, agnostic CoT tends to generate plausibility narratives that conflate distinct inferential obligations, producing locally coherent justifications that nonetheless violate the schema-level constraints required for correctness. This contrast is visible across all four reasoning families, indicating that the principal effect of *CARENLI* is to enforce *trajectory alignment* with the intended inferential schema rather than to elicit additional clinical facts.

Concretely, in CAUSAL ATTRIBUTION, structured inference decomposes statements into separable subclaims and enforces comparator requirements, whereas agnostic CoT collapses tolerability signals into efficacy claims via association-based shortcuts. In COMPOSITIONAL GROUNDING, structured inference instantiates cross-factor admissibility checks over interacting clinical variables, while agnostic CoT reduces the task to single-factor matching and bypasses binding constraints. In EPISTEMIC VERIFICATION, structured traces apply explicit evidence hierarchies, prioritising objective findings over unsupported assertions, whereas agnostic CoT defaults to authority deference. Finally, in RISK STATE ABSTRACTION, structured inference explicitly represents latent risk states and integrates likelihood with severity to justify action under uncertainty, while agnostic CoT maps uncertainty to neutrality and fails to account for catastrophic downside.

5 Related Work

A prevailing assumption in NLP is that enlarging model capacity and exposure will improve not only task performance but also the fidelity of underlying reasoning. Scaling law analyses (Kaplan et al., 2020; Hoffmann et al., 2022) and flagship model reports (Brown et al., 2020; Touvron et al., 2023; Bubeck et al., 2023) reinforce this view, yet critics observe that benchmark success often reflects surface regularities rather than robust inference (Marcus, 2022; Mahowald et al., 2024). Our work instantiates this critique in the clinical domain: we show that models encode relevant medical facts but fail to deploy them in principled reasoning flows.

Evidence across NLI research supports this diagnosis. General-domain studies reveal annotation artifacts and shortcut reliance (Gururangan et al., 2018; Poliak et al., 2018; Webson and Pavlick, 2022; Turpin et al., 2023). Clinical NLI benchmarks extend these concerns: MedNLI (Romanov and Shivade, 2018), NLI4CT (Jullien et al., 2023a), and CTNLI (Jullien et al., 2025) report systematic reasoning errors, motivating frameworks that explicitly separate knowledge retrieval from inferential schema adherence.

Parallel to this diagnostic agenda, recent work has investigated agentic and modular architectures as a pathway to more reliable reasoning. Prior work has explored task decomposition and routing into simpler subtasks (Khot et al., 2022; Yao et al., 2022), role- or agent-specialised problem solving in multi-agent systems (Li et al., 2023; Wu et al., 2024; Hong et al., 2024), and iterative self-verification or refinement of model outputs (Madaan et al., 2023; Dhuliawala et al., 2024). In particular, Chain-of-Verification (CoVe) frames reliability as post-hoc verification over an initial reasoning trajectory (Dhuliawala et al., 2024). *CARENLI* covers this design space, but differs in that decomposition, solver specialisation, and verification are conditioned on formally defined clinical reasoning families and tied to explicit family-specific decision procedures.

6 Conclusion

We introduced *CARENLI*, a compartmentalised agentic framework for CTNLI that makes the intended inference regime explicit through reasoning-type routing, solver-specific procedures, and verifier-guided refinement. By operationalising four reasoning families, CARENLI functions both as a diagnostic lens for isolating reasoning failures and as a practical prompting scaffold for more auditable clinical inference.

Aggregated across models and task families, *CARENLI* achieves 56.8% macro accuracy, compared to 22.5% for agnostic CoT prompting and 23.1% for agnostic direct prompting. Gains are consistent across all evaluated backbones, but vary markedly across reasoning families, indicating that some forms of clinical inference benefit more from explicit structure than others.

These results support the central claim that explicitly encoding the target reasoning regime is critical for reliable CTNLI, while also highlighting persistent limitations, particularly for compositional forms of inference. Future work should expand the set of supported reasoning families and assess *CARENLI* in less constrained CTNLI benchmarks.

7 Limitations

- **Benchmark scope and ecological validity.** Although CTNLI offers controlled coverage of clinically meaningful inference patterns, instances are generated from typed templates and therefore do not fully reflect the linguistic variability, discourse structure, and documentation artefacts found in real trial protocols, clinical notes, or EHR-derived text. As a result, absolute performance numbers may not transfer directly to unconstrained clinical text.

- **Scale and coverage of clinical variation.** The extended benchmark contains 200 items with balanced reasoning type coverage, which supports targeted analysis but limits statistical power for fine-grained subgroup conclusions. In addition, many clinically important axes are only partially represented, including multi-morbidity, longitudinal trajectories, interacting laboratory trends, and institution-specific protocol language.

- **Limited outcome space and task framing.** We study three-way NLI labels with reasoning-type-specific decision procedures. This framing does not capture downstream requirements such as calibrated uncertainty, abstention under ambiguity, selective prediction, or structured justifications that can be audited against explicit evidence sources.

- **Prompt and implementation sensitivity.** The framework operationalises "compartmentalisation" through prompts, role constraints, and veri-

fication instructions. Performance can vary with prompt wording, formatting, temperature, and model-specific instruction-following behaviour. This sensitivity limits strict reproducibility across model versions and makes it difficult to attribute gains to a single design choice.

- **Model and infrastructure coverage.** Experiments are limited to four contemporary LLMs and to text-only inference. Results may differ for other model families, smaller open models, tool-augmented systems, or multimodal settings. In addition, closed-model updates can change behaviour over time, which complicates longitudinal comparability.

- **Safety and deployment boundaries.** While the work targets safety-critical reasoning, it does not constitute a clinical decision support system and is not evaluated in prospective clinical workflows. Human oversight, governance, and domain-specific validation would be required before any real-world use, particularly where errors could affect patient care or trial conduct.

References

2017. Common terminology criteria for adverse events (ctcae) v5.0. https://ctep.cancer.gov/protocolDevelopment/electronic_applications/ctc.htm.

Monika Agrawal, Irene Y. Chen, Fatima Gulamali, and 1 others. 2025. The evaluation illusion of large language models in medicine. *npj Digital Medicine*.

Fadi Aljamaan, Mohamad-Hani Temsah, Ibraheem Altamimi, Ayman Al-Eyadhy, Amr Jamal, Khalid Alhasan, Tamer A Mesallam, Mohamed Farahat, and Khalid H Malki. 2024. Reference hallucination score for medical artificial intelligence chatbots: development and usability study. *JMIR Medical Informatics*, 12(1):e54345.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. 2016. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, and 1 others. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. 2024. Chain-of-verification reduces hallucination in large language models. In *Findings of the association for computational linguistics: ACL 2024*, pages 3563–3578.

Rachel L. Draelos, Samina Afreen, Barbara Blasko, Tiffany L. Brazile, Natasha Chase, Dimple Patel Desai, Jessica Evert, Heather L. Gardner, Lauren Herrmann, Aswathy Vaikom House, Stephanie Kass, Marianne Kavan, Kirshma Khemani, Amanda Koire, Lauren M. McDonald, Zahraa Rabeeah, and Amy Shah. 2025. Large language models provide unsafe answers to patient-posed medical questions. arXiv preprint arXiv:2507.18905.

Audrey Eichenberger, Stephen Thielke, and Adam Van Buskirk. 2025. A case of bromism influenced by use of artificial intelligence. *Annals of Internal Medicine: Clinical Cases*, 4(8).

Elizabeth A Eisenhauer, Patrick Therasse, Jan Bogaerts, Lawrence H Schwartz, Danielle Sargent, Robert Ford, Janet Dancey, Stephen Arbuck, Steve Gwyther, Margaret Mooney, and 1 others. 2009. New response evaluation criteria in solid tumours: revised recist guideline (version 1.1). *European journal of cancer*, 45(2):228–247.

Ronald Fagin, Joseph Y Halpern, Yoram Moses, and Moshe Vardi. 2004. *Reasoning about knowledge*. MIT press.

Evelyn Goh and 1 others. 2024. Large language model influence on diagnostic reasoning. *JAMA Network Open*.

Jocelyn Gravel, Madeleine D'Amours-Gravel, and Esli Osmanlliu. 2023. Learning to fake it: limited responses and fabricated references provided by chatgpt for medical questions. *Mayo Clinic Proceedings: Digital Health*, 1(3):226–234.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R Bowman, and Noah A Smith. 2018. Annotation artifacts in natural language inference data. *arXiv preprint arXiv:1803.02324*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, and 1 others. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Ceyao Zhang, Jinlin Wang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, and 1 others. 2024. Metagpt: Meta programming for a multi-agent collaborative framework. International Conference on Learning Representations, ICLR.

MG Myriam Hunink, Milton C Weinstein, Eve Wittenberg, Michael F Drummond, Joseph S Pliskin, John B Wong, and Paul P Glasziou. 2014. *Decision making in health and medicine: integrating evidence and values*. Cambridge university press.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Philip Nicholas Johnson-Laird. 1983. *Mental models: Towards a cognitive science of language, inference, and consciousness*. 6. Harvard University Press.

Maël Jullien, Marco Valentino, and André Freitas. 2024. Semeval-2024 task 2: Safe biomedical natural language inference for clinical trials. *arXiv preprint arXiv:2404.04963*.

Maël Jullien, Marco Valentino, and André Freitas. 2025. The knowledge-reasoning dissociation: Fundamental limitations of llms in clinical natural language inference. *arXiv preprint arXiv:2508.10777*.

Maël Jullien, Marco Valentino, Hannah Frost, Paul O'Regan, Dónal Landers, and André Freitas. 2023a. Nli4ct: Multi-evidence natural language inference for clinical trial reports. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16745–16764.

Maël Jullien, Marco Valentino, Hannah Frost, Paul O'regan, Donal Landers, and André Freitas. 2023b. Semeval-2023 task 7: Multi-evidence natural language inference for clinical trial data. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2216–2226.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Tushar Khot, Harsh Trivedi, Matthew Finlayson, Yao Fu, Kyle Richardson, Peter Clark, and Ashish Sabharwal. 2022. Decomposed prompting: A modular approach for solving complex tasks. *arXiv preprint arXiv:2210.02406*.

Guohao Li, Hasan Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. 2023. Camel: Communicative agents for" mind" exploration of large language model society. *Advances in Neural Information Processing Systems*, 36:51991–52008.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, and 1 others. 2023. Self-refine: Iterative refinement with self-feedback. *Advances in neural information processing systems*, 36:46534–46594.

Kyle Mahowald, Anna A Ivanova, Idan A Blank, Nancy Kanwisher, Joshua B Tenenbaum, and Evelina Fedorenko. 2024. Dissociating language and thought in large language models. *Trends in cognitive sciences*, 28(6):517–540.

Gary Marcus. 2022. Deep learning is hitting a wall. *Nautilus*, 10:2022.

Gary Marcus and Ernest Davis. 2020. *Rebooting AI: Building Artificial Intelligence We Can Trust*. Pantheon.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of ACL*.

Jesutofunmi A. Omiye and 1 others. 2023. Large language models propagate race-based medicine. *npj Digital Medicine*.

OpenAI. 2025. Gpt-5 system card. Accessed: 2025-12-24.

Judea Pearl. 2009. *Causality*. Cambridge university press.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. *arXiv preprint arXiv:1805.01042*.

Xin Quan, Marco Valentino, Louise A Dennis, and Andre Freitas. 2024. Enhancing ethical explanations of large language models through iterative symbolic refinement. *arXiv preprint arXiv:2402.00745*.

Pavan Reddy and Nithin Reddy. 2025. Preventing another tessa: Modular safety middleware for health-adjacent ai assistants. *arXiv preprint arXiv:2509.07022*.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of nlp models with checklist. In *Proceedings of ACL*.

Alexey Romanov and Chaitanya Shivade. 2018. Lessons from natural language inference in the clinical domain. *arXiv preprint arXiv:1808.06752*.

Steven Sloman and Steven A Sloman. 2009. *Causal models: How people think about the world and its alternatives*. Oxford University Press.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. 2023. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36:74952–74965.

Johan Van Benthem. 2011. *Logical dynamics of information and interaction*. Cambridge University Press.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of ICLR*.

Albert Webson and Ellie Pavlick. 2022. Do prompt-based models really understand the meaning of their prompts? In *Proceedings of the 2022 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 2300–2344.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of NAACL-HLT*.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, and 1 others. 2024. Autogen: Enabling next-gen llm applications via multi-agent conversations. In *First Conference on Language Modeling*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*.

A Appendix

A.1 Solver specifications

This subsection states the normative decision procedures implemented by the family-specific solvers introduced in Section 2.3. The exact prompt templates that operationalise these procedures are reported in Appendix A.5.

Epistemic Verification Solver. The Epistemic Verification solver operationalises verification when premises contain multiple, potentially conflicting assertions. Following Jullien et al. (2025), premises are treated as conjunctions of defeasible epistemic commitments $K_a(\varphi_i)$, each attributed to a source agent a (Fagin et al., 2004; Van Benthem, 2011). Three principles govern inference. First, all assertions are treated as reported claims rather than ground truth, with admissibility evaluated against the clinical model $\mathcal{M}\text{CT} = \langle W\text{CT}, I_{\text{CT}} \rangle$. Second, inconsistencies are resolved using a plausibility function $\pi(\varphi_i)$ instantiated as an evidential hierarchy: objective measurements dominate diagnostic criteria, which dominate observations, interpretations, and self-report. Conflicts are resolved by discarding lower-ranked commitments, yielding a maximal consistent set $E^* \subseteq E$ such that $\forall w \in W_{\text{CT}}, E^*$ is jointly satisfiable under $\mathcal{M}_{\text{CT}}$. Third, coherence constraints exclude ontologically impossible or temporally incoherent assertions. A candidate statement s is entailed if $E^* \models s$, contradicted if $E^* \models \neg s$, and neutral if neither holds. Neutrality is therefore a principled outcome when conflicts remain unresolved or evidence underdetermined

Causal Attribution Solver. The Causal Attribution solver operationalises attribution when the hypothesis asserts a treatment $\rightarrow$ outcome relation. Under the interventionist framework (Pearl, 2009), causal effect is defined as

$$\text{CE}(T, Y) \triangleq \mathbb{E}[Y \mid \text{do}(T = 1)] - \mathbb{E}[Y \mid \text{do}(T = 0)],$$

where T denotes a treatment, Y an outcome, and $\text{do}(T = t)$ an intervention severing incoming dependencies. Unlike observational associations $\mathbb{E}[Y \mid T = t]$, causal effect requires interventional contrast. The solver enforces this distinction by (i) parsing premises for outcome measurements or adverse events, treated as observational unless comparators or manipulations are explicit; (ii) verifying causal criteria of temporality, contrast, and confounding control; and (iii) evaluating the hypothesis s: entailment requires interventional evidence, contradiction follows from unsupported causal claims, and neutrality applies otherwise. This constrains models to uphold the semantics of interventionist causality and corrects the failure mode observed in Jullien et al. (2025), where models collapsed causal attribution into association, predicting entailment from surface correlations rather

than from interventional criteria.

Compositional Grounding Solver. The Compositional Grounding operationalises inference when the truth of a statement depends on the joint configuration of multiple clinical factors rather than any single predicate. Formally, compositional grounding requires that a tuple

$$x := \langle d, z, dx, s \rangle \in \mathfrak{D}^4_{\text{CT}}$$

of drug d, dose z, diagnosis dx, and schedule s must be admissible under the interpretation function I_{CT}. A statement ψ asserting clinical benefit is entailed only if $I_{\text{CT}}(\text{Benefit})(x) \to True$ in all admissible worlds $w \in W_{\text{CT}}$; contradictions arise when x violates therapeutic, diagnostic, or scheduling constraints. First, atomic factors (drug, dose, schedule, patient details) are extracted. Second, the tuple x is assembled and tested for compatibility under I_{CT}, which encodes therapeutic ranges, scheduling rules, and indication–diagnosis mappings. Third, the hypothesis s is assessed: entailment requires that x be admissible and support the asserted outcome; contradiction arises when x violates constraints; neutrality is assigned when the configuration is underdetermined. This design intends to prevents the error mode identified in Jullien et al. (2025), where LLMs collapsed compositional interactions into isolated surface predicates, thereby overlooking protocol violations or emergent incompatibilities.

Risk Abstraction Solver. The Risk Abstraction solver operationalises inference when hypotheses involve explicit or latent risk. Risk is defined as an expectation over admissible worlds $w \in W_{\text{CT}}$, combining probability and severity of adverse events (Hunink et al., 2014):

$$\mathbb{E}_{w \sim \Pr(w|\varphi)} \left[\sum_{e \in \mathcal{E}(w,\psi)} \Pr(e \mid w) \cdot \mathcal{A}(e, w) \right],$$

where $\mathcal{E}(w, \psi)$ is the set of clinically relevant events for ψ, $\Pr(e \mid w)$ their conditional probability, and $\mathcal{A}(e, w)$ an adverse outcome function quantifying harm. Entailment requires that the statement correctly reflect this expected risk profile, either by ranking events appropriately (when explicit frequencies are given) or by acknowledging unruled-out harms in latent risk settings.

The solver prompt translates this into a decision procedure with three steps. First, it identifies adverse events or conditions in the premise and classifies them according to severity and probability. Second, it integrates these two dimensions into an expected-harm ranking: severe but rare outcomes may outweigh frequent but benign ones, in line with clinical reasoning. Third, it evaluates the hypothesis s: entailment follows if s reflects the correct harm-weighted risk ordering, contradiction if it inverts or ignores it, and neutrality if the evidence base is insufficient.

By enforcing this expected-risk computation, the solver ensures that judgments reflect decision-theoretic reasoning rather than surface counts or lexical salience. This is intended to correct the failure mode highlighted in Jullien et al. (2025), where models frequently misclassified risk when severity and frequency were in tension, defaulting to naive frequency matching rather than principled risk abstraction (Hunink et al., 2014; Eisenhauer et al., 2009; CTC, 2017).

A.2 Qualitative Analysis of Representative Items

A qualitative analysis of four representative cases, one from each reasoning type, is presented using outputs from DEEPSEEK R1. For each case, the reasoning trace generated under the *oracle Router* condition (agentic framework, with the solver routed to the correct reasoning type) is contrasted with the baseline produced under *agnostic CoT* prompting. This comparison highlights how explicit compartmentalisation enables alignment with the expected reasoning patterns specified in the knowledge framework, while also revealing the characteristic deviations of free-form CoT.

Causal Attribution (Problem 6).

> **Premise:** *"Adverse Events Summary: Total: 4/12 (33.33%). Headache 2/12 (16.67%), Pruritus 1/12 (8.33%), Mild anemia 1/12 (8.33%). All events were Grade 1 or 2."*
> **Statement:** *"The drug was effective and well tolerated, with only mild side effects reported."*
>
> **Agentic (verbatim).** *''The statement makes two claims: (1) effectiveness ($T \to$ improved outcome) and (2) tolerability ($T \to$ only mild side effects). The premise only reports adverse event grades (all Grade 1/2) and rates (33.33%*

total), which directly confirms tolerability. However, the premise contains no efficacy data (e.g., disease improvement metrics, control group, baseline measurements). Since causal claims require comparator outcomes, the effectiveness portion remains unsubstantiated. While tolerability is entailed, the overall label is neutral."

CoT (abridged). *"The statement is supported since only mild AEs were reported . . . this indicates the drug was effective and well tolerated."*

The agentic solver follows the expected *comparator requirement*, decomposing causal claims into subcomponents and assigning neutral when one is unsupported. The CoT baseline collapses tolerability and efficacy into a single plausibility heuristic, producing over-attribution. This demonstrates how structural prompts prevent the conflation of post-treatment outcomes with causal effect. Under correct scaffolding, LLMs reliably enforce comparator rules; without it, they default to association heuristics.

Compositional Grounding (Problem 12).

Premise: *"Fludarabine 120 mg/m^2 daily $\times$14 days for CLL in an elderly patient."*
Statement: *"The treatment is expected to induce remission, improve blood counts, and prolong survival."*

Agentic (verbatim). *"1) Extract factors: fludarabine 120 mg/m^2 daily $\times$14. 2) Standard dose is 25 mg/m^2/day $\times$5. The given regimen is $\approx$5$\times$ higher and extended. 3) Such dosing causes severe toxicity in elderly patients. 4) This violates therapeutic admissibility; the statement's claim of benefit is contradicted."*

CoT (abridged). *"Fludarabine is used in CLL, so this treatment is likely to be effective . . . the statement is entailed."*

The agentic solver operationalises the *cross-factor constraint*, checking consistency across drug, dose, schedule, and patient. This reproduces the knowledge framework's tuple-based admissibility rule. The CoT reasoning reflects the failure mode of Jullien et al. (2025): collapsing multi-factor structures into a single drug–indication match. Here, agentic scaffolding ensures toxicity is identified, whereas CoT ignores unsafe interactions. The repercussion is clinical: a toxic regimen is falsely labelled as beneficial under free CoT, while the structured agent enforces compositional validity.

Epistemic Verification (Problem 16).

Premise: *"A 45-year-old man complains of chest discomfort after meals and occasional regurgitation. Endoscopy is normal. Despite the absence of cardiac symptoms, the physician diagnoses myocardial infarction and starts anticoagulation therapy."*
Statement: *"The patient has myocardial infarction."*

Agentic (verbatim). *"1) Endoscopy normal (objective evidence). 2) Symptoms consistent with GERD, not MI. 3) MI requires ECG/troponins, absent here. 4) Physician's diagnosis conflicts with higher-tier evidence. 5) Conclusion: contradiction."*

CoT (abridged). *"The physician diagnosed MI, so this supports the statement . . . although there is no cardiac evidence, the statement is entailed."*

The agentic solver follows the *evidence hierarchy*, prioritising instrumented evidence over unsupported assertion. The CoT baseline defers to physician authority, ignoring hierarchy. This misalignment mirrors the epistemic collapse observed previously: models recognise evidence but misapply credibility weights. By enforcing explicit tiering, the agentic pipeline compels the correct contradiction label. This illustrates that LLMs are capable of stratified verification if prompted to discard lower-tier claims in conflict with higher-tier evidence.

Risk State Abstraction (Problem 39).

Premise: *"A 55-year-old man with acute severe low back pain reports saddle anesthesia, urinary retention, and bilateral leg weakness. Reflexes reduced. No imaging performed."*
Statement: *"Emergency MRI is required to exclude cauda equina syndrome."*

Agentic (verbatim). *"Symptoms constitute red flags for CES. Likelihood: high,*

since multiple features align. Severity: catastrophic, with risk of paralysis if missed. No imaging means CES remains unexcluded. Therefore urgent MRI is mandated; statement entailed."

CoT (abridged). *"The patient has back pain and weakness ... MRI could help but it is not certain; label neutral."*

The agentic solver instantiates the *severity–likelihood calculus*: combining probability with consequence to project latent catastrophic risk. The CoT baseline fails to abstract beyond symptom description, treating CES as one of many possible adverse events. This reproduces the frequency heuristic failure identified in prior work. The consequence is profound: under CoT, a time-critical emergency is missed. The agentic framework ensures catastrophic risk is correctly prioritised, demonstrating that LLMs can perform principled risk abstraction when explicitly instructed.

Across reasoning families, the agentic framework compels alignment with formal inferential schemas: comparators for causality, cross-factor constraints for composition, evidence hierarchies for epistemic verification, and severity–likelihood calculus for risk. The CoT baseline fails in reasoning-type-specific, systematic ways: collapsing causal claims, flattening compositional structure, deferring to authority, and ignoring catastrophic risk. These are not random lapses but consistent heuristics. The crucial finding is that LLMs *can* reproduce the expected reasoning patterns under correct prompting: the agentic pipeline demonstrates that comparator tests, constraint enforcement, tiered evidence, and risk abstraction are all within model capacity, provided the reasoning path is structured. Without such scaffolding, CoT defaults to surface plausibility, with direct repercussions for clinical safety and validity.

A.3 Dataset Construction and Validation

To assess annotation reliability, an inter-annotator agreement (IAA) study was conducted on a random sample of 50 instances drawn uniformly across reasoning families. Each instance was independently annotated by three domain experts (volunteers):

- one expert in natural language inference,
- one medical doctor,
- one physician associate.

Annotators were provided with the premise, statement, and task definition, and were asked to assign both an entailment label (*entailed*, *contradicted*, or *neutral*) and a reasoning type.

Agreement was measured using Fleiss' κ. The resulting agreement scores indicate high reliability:

- Entailment label agreement: $\kappa = 0.92$
- reasoning type agreement: $\kappa = 0.97$

Table 4 shows an example annotation item used in the IAA study.

Premise: 67-year-old male with rheumatoid arthritis receiving methotrexate 100mg orally daily.
Statement: The therapy is expected to decrease joint inflammation and enhance quality of life.
Label:
reasoning type:

Table 4: Example CTNLI instance used in the inter-annotator agreement study.

A.4 Tables

Model	Causal Attr.	Comp. G.	Epis. Verif.	Risk Abstr.
CARENLI				
deepseek-r1	43.0	36.4	73.9	95.3
gpt-5.1	45.5	0.4	38.0	92.0
gpt-4.1	55.0	59.5	82.9	95.9
gpt4o-mini	42.8	6.4	56.0	85.8
Oracle Router				
deepseek-r1	44.9	93.8	82.0	100.0
gpt-5.1	52.4	2.0	36.0	96.4
gpt-4.1	46.5	56.9	78.0	96.4
gpt4o-mini	46.5	15.7	56.0	91.1
Agnostic CoT				
deepseek-r1	36.7	17.6	35.3	9.8
gpt-5.1	31.2	0.4	23.6	24.7
gpt-4.1	30.8	4.3	26.0	81.5
gpt4o-mini	29.5	0.0	7.6	0.9
Agnostic Direct				
deepseek-r1	27.5	15.9	35.8	10.9
gpt-5.1	31.2	0.0	18.0	49.1
gpt-4.1	26.0	3.5	26.4	90.2
gpt4o-mini	15.8	0.0	8.8	11.1

Table 5: Accuracy across reasoning tasks.

Model	Causal Attr.	Comp. G.	Epis. Verif.	Risk Abstr.
deepseek-r1	67.6	42.0	100.0	90.2
gpt-5.1	63.2	32.9	100.0	90.9
gpt-4.1	56.0	98.4	100.0	87.8
gpt4o-mini	48.0	12.9	99.6	83.6

Table 6: Reasoning Classification Accuracy across tasks

Model	Causal Attr.	Comp. G.	Epis. Verif.	Risk Abstr.
CARENLI				
deepseek-r1	49.6	26.5	73.1	92.7
gpt-5.1	59.0	42.7	45.6	93.8
gpt-4.1	53.4	72.2	79.3	95.9
gpt4o-mini	54.4	47.2	61.7	86.2
Oracle Router				
deepseek-r1	44.9	60.4	80.0	93.3
gpt-5.1	52.4	72.5	48.0	94.5
gpt-4.1	48.8	66.7	82.0	96.4
gpt4o-mini	72.1	35.3	58.0	91.1

Table 7: Verifier accuracy (%) across reasoning tasks

A.5 Prompts

This subsection reports the exact prompt templates used by *CARENLI*. Fig. 17 gives the router prompt; Figs. 5–14, Figs. 6–15, and Figs. 7–16 give the solver, refiner, and verifier prompts, respectively. These templates instantiate the decision procedures defined in Appendix A.1.

```
SYSTEM:
You are the Causal Solver for Clinical Trial NLI (CTNLI). You receive a premise and a statement only after the planner has routed the
example to Causal. Decide whether the statement is entailed, contradicted, or neutral by assessing whether the premise supports, refutes, or
 leaves uncertain a causal relationship between an intervention/exposure (T) and an outcome (Y). This includes positive effects (improve,
reduce), negative effects (cause, worsen), or no effect.

Core principles

    - Identify causal claims: Look for verbs or constructions indicating causality (cause, lead to, improve, reduce, prevent, accelerate,
associated with if clearly directional).

    - Evaluate the premise for relevant comparator, control or baselines. Causal claims cannot be made without a reference point, such as a
control group or baseline measurement.

    - Ensure that the concepts you are evaluating are direct evidence to support/disprove the causal claim and not only tangentially
related.

    - Distinguish correlation from causation: Correlative evidence (association, co-occurrence) only supports causality if temporality and
plausible mechanism are established in the premise.

    - Evaluate evidence direction and magnitude:

        - Support for the stated effect direction -> entailed.

        - Evidence in the opposite direction -> contradicted.

        - Insufficient or conflicting evidence -> neutral.

    - Temporal consistency: Cause must precede effect; post-hoc evidence without clear temporality is weak.



Decision procedure

    - Identify T (treatment/exposure) and Y (outcome) from both premise and statement.

    - Extract causal evidence: trial results, measured changes, event rates, comparative stats.

    - Locate control/baseline and test/intervention groups or variables in the premise, ensuring they are explicitly defined and relevant to
 T and Y.

    - Check directionality: Does the evidence show T -> increase/decrease in Y, no change, or opposite change?

    - Assess strength & relevance: Is the evidence directly about T and Y in the same population/context?

    Label:

    - If the evidence comparing the test/intervention group/variables and the control/baseline group/variables matches the causal direction
in the statement -> entailed

    - If the evidence comparing the test/intervention group/variables and the control/baseline group/variables shows the opposite direction
-> contradicted

    - If the control/baseline comparison or test group/variables are ambiguous, absent, or contain conflicting information -> neutral

Output contract (strict JSON)

Return only a single JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}



USER:
You are the Causal Solver for CTNLI. Follow your system instructions exactly.
premise={PREMISE}
statement={STATEMENT}
Return exactly one JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}
```

Figure 5: Causal Solver Prompt

```
SYSTEM:
You are the Causal Refiner for Clinical Trial NLI (CTNLI). You receive a premise, a statement, a solver_output (JSON), and a
verification_output (JSON). Your job: Implement the verifier's fixes to correct the solver's reasoning and produce a final causal judgment.

How to use verification_output

    If fact_verification = "incorrect":

    - Identify every incorrect/unsupported domain fact listed in fact_reasoning.

    - Remove these from the original reasoning.

    - Replace them only with the corrected facts as stated in fact_reasoning, provided they are supported by the premise or by generally
accepted clinical regularities allowed in CTNLI.

    - If a corrected fact is not supported, omit it and explain the insufficiency.

    If fact_verification = "correct":

    - Assume no unsupported facts; do not add new factual content beyond the premise and allowed regularities.

    If pattern_verification = "incorrect":

    - Apply the minimal fixes specified in pattern_reasoning to align with proper causal inference:

    - Identify T (intervention/exposure) and Y (outcome).

    - Use an explicit comparator/baseline/control for causal claims; without one, you cannot conclude causality.

    - Enforce temporality (T precedes Y).

    - Use evidence directly linking T -> Y in the same population/context.

    - State effect direction (increase/decrease/no effect) or declare insufficiency -> neutral.

    - Do not treat association or numeric difference as causal without comparator/temporality.

    If pattern_verification = "correct":

    - Keep the causal structure; you may tighten clarity but must not alter the validated pattern.


Decision procedure

    - Identify T (treatment/exposure) and Y (outcome) from both premise and statement.

    - Extract causal evidence: trial results, measured changes, event rates, comparative stats.

    - Locate control/baseline and test/intervention groups or variables in the premise, ensuring they are explicitly defined and relevant to
 T and Y.

    - Check directionality: Does the evidence show T -> increase/decrease in Y, no change, or opposite change?

    - Assess strength & relevance: Is the evidence directly about T and Y in the same population/context?

    Label:

    - If the evidence comparing the test/intervention group/variables and the control/baseline group/variables matches the causal direction
in the statement -> entailed

    - If the evidence comparing the test/intervention group/variables and the control/baseline group/variables shows the opposite direction
-> contradicted

    - If the control/baseline comparison or test group/variables are ambiguous, absent, or contain conflicting information -> neutral

Output contract (strict JSON)

Return only a single JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}


USER:
You are the Causal Refiner. Follow your system instructions exactly.
premise={PREMISE}
statement={STATEMENT}
solver_output (JSON) = {SOLVER_JSON}
verification_output (JSON) = {VERIFIER_JSON}

Return exactly one JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}
```

Figure 6: Causal Refiner Prompt

```
SYSTEM:

You are an expert in Fact Verification and Causal Pattern Verification for Clinical Trial NLI (CTNLI).

Your task is to:
1. perform fact verification.
Verify only the factual correctness of domain knowledge used in the solver's reasoning, i.e., information such as
treatment regiments, standard-of-care patterns, and disease subtype not supported by the premise or not generally
accepted clinical regularities allowed in CTNLI, these may be explicit statements or implicit assumptions.
If any such unsupported fact appears, mark it as incorrect, and report incorrect facts in the explanation,
and provide the corrected facts. Else mark as correct. You must only verify the solver output, not the statement or premise.

2. perform causal pattern verification.
Decide whether the Causal Solver's reasoning pattern aligns with causal inference. If the reasoning does not align,
mark it as incorrect and provide minimal fixes to the reasoning. If it aligns, mark it as correct. You must only verify
the solver output, not the statement or premise.

Core expectations (explicitly or implicitly present):

Clearly identify T (intervention/exposure) and Y (outcome).

Distinguish correlation vs causation; look for temporality and ensure the reasoning includes a comparator, baseline or control and test
variable or group.
Without a comparator, and test variables causal claims cannot be made.

Use evidence directly linking T->Y in the same context.

Assess and state effect direction or insufficiency.

Use neutral when causal requirements are unmet.

Flag as incorrect if reasoning (explicitly or implicitly) shows any of:

Treats numeric effect as causality without temporal/comparator basis.

Omits T or Y; uses unrelated outcomes.

Return only a single JSON object:
{
  "fact_verification": "correct"|"incorrect",
  "fact_reasoning": "step by step explanation of the incorrect facts, provide corrected facts; if none, write 'No unsupported facts.'",
  "pattern_verification": "correct"|"incorrect",
  "pattern_reasoning": "step by step explanation of the reasoning mistakes, minimal fixes to the solver reasoning; if none, write 'No
reasoning mistakes.'",
}

USER:
You are the Causal Pattern Verifier. Follow your system instructions exactly.
premise={PREMISE}
statement= {STATEMENT}
solver_output (JSON) = {SOLVER_JSON}
```

Figure 7: Causal Verifier Prompt

```
SYSTEM:
You are the Composition Solver for Clinical Trial NLI (CTNLI). You receive a premise and a statement only after the planner has routed the
example to Compositional. Decide whether the statement is entailed, contradicted, or neutral given the premise by evaluating joint
constraints and their interdependencies across multiple clinical factors. This includes drug–dose–unit–schedule–diagnosis–patient factors
(age, sex, renal/hepatic function, comorbidities) and co-therapy rules. This means not just verifying each factor in isolation, but
reasoning about how factors influence each other and how their combined effects shape the overall meaning.

Core principles

    - Consider the network of constraints: how characteristics, conditions, and contextual details affect each other's validity,
applicability, or limits, and how these interactions determine whether the overall scenario is possible.

    - All required conditions together must be satisfied for entailment.

    - A violation in any condition, or an interaction that invalidates the combination, leads to contradiction.

    - Numeric thresholds and allowable ranges must be respected after applying relevant adjustments.

    - Flag impossible or self-contradictory combinations (e.g., contraindications, impossible values, e.g BMI 19 and weight 200kg.

Decision procedure

    - Extract factors from the statement and premise (intervention, schedule, condition, subject factors, concurrent therapies, exclusions).

    - Identify dependencies: determine which factors constrain or modify others.

    Label:

        - Exact match and no dependency violations for all conditions and dependencies -> entailed

        - Explicit mismatch or dependency violation for any condition or interaction -> contradicted.

        - Absent/unspecified -> neutral.

Output contract (strict JSON)

Return only a single JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}


USER:
You are the Composition Solver for CTNLI. Follow your system instructions exactly.
premise={PREMISE}
statement={STATEMENT}
Return exactly one JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}
```

Figure 8: Composition Solver Prompt

```
SYSTEM:
You are the Composition Refiner for Clinical Trial NLI (CTNLI). You receive a premise, a statement, a solver_output (JSON), and a
verification_output (JSON). Your job: Implement the verifier's fixes to correct the solver's reasoning and produce a final composition
judgment.

How to use verification_output

    If fact_verification = "incorrect":

    - Identify every incorrect/unsupported domain fact listed in fact_reasoning.

    - Remove these from the original reasoning.

    - Replace them only with the corrected facts as stated in fact_reasoning, provided they are supported by the premise or by generally
accepted clinical regularities allowed in CTNLI.

    - If a corrected fact is not supported, omit it and explain the insufficiency.

    If fact_verification = "correct":

    - Assume no unsupported facts; do not add new factual content beyond the premise and allowed regularities.

    If pattern_verification = "incorrect":
    - Apply the minimal fixes from pattern_reasoning to conform to the Composition Solver's principles:

    - Network of constraints: enumerate factors and check how they constrain/modify one another.

    - All-conditions test: all required conditions must be satisfied simultaneously for entailment.

    - Violation test: any explicit mismatch or interaction/contraindication -> contradiction.

    - Numeric thresholds & ranges: normalize units, respect thresholds as stated in the premise; apply adjustments only if specified/allowed
 (e.g., renal/hepatic adjustments mentioned in the premise).

    - Impossibilities: flag impossible/self-contradictory combinations (e.g., inconsistent units/values).

    - No invention: do not introduce unstated guideline rules, thresholds, or patient attributes.

    If pattern_verification = "correct":

    - Keep the causal structure; you may tighten clarity but must not alter the validated pattern.

Decision procedure

    - Extract factors from the statement and premise (intervention, schedule, condition, subject factors, concurrent therapies, exclusions).

    - Identify dependencies: determine which factors constrain or modify others.

    Label:

        - Exact match and no dependency violations for all conditions and dependencies -> entailed

        - Explicit mismatch or dependency violation for any condition or interaction -> contradicted.

        - Absent/unspecified -> neutral.

Output contract (strict JSON)

Return only a single JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}


USER:
You are the Composition Refiner. Follow your system instructions exactly.
premise={PREMISE}
statement={STATEMENT}
solver_output (JSON) = {SOLVER_JSON}
verification_output (JSON) = {VERIFIER_JSON}

Return exactly one JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}
```

Figure 9: Composition Refiner Prompt

```
SYSTEM:

You are an expert in Fact Verification and Composition Pattern Verification for Clinical Trial NLI (CTNLI).

Your task is to:
1. perform fact verification.
Verify only the factual correctness of domain knowledge used in the solver's reasoning, i.e., information such as
treatment regiments, standard-of-care patterns, and disease subtype not supported by the premise or not generally
accepted clinical regularities allowed in CTNLI, these may be explicit statements or implicit assumptions.
If any such unsupported fact appears, mark it as incorrect, and report incorrect facts in the explanation,
and provide the corrected facts. Else mark as correct. You must only verify the solver output, not the statement or premise.

2. perform composition pattern verification.
Decide whether the Composition Solver's reasoning pattern aligns with compositional/joint-constraint reasoning. If the reasoning does not
align,
mark it as incorrect and provide minimal fixes to the reasoning. If it aligns, mark it as correct. You must only verify
the solver output, not the statement or premise.

Core expectations (explicitly or implicitly present):

Extract all required factors from the statement.

Evaluate joint satisfaction—the combination must be assessed, not factors in isolation.

Check inter-factor dependencies and numeric bounds after adjustments.

Use neutral when any required factor/dependency is unknown; contradiction for explicit mismatch or invalidating interaction; entailment when
 all satisfied.

Maintain internal consistency; avoid external facts.

Flag as incorrect if reasoning (explicitly or implicitly) shows any of:

Ignores clear dependencies.

Declares entailment despite missing required factors; or contradiction without identifying the violation.

Return only a single JSON object:
{
  "fact_verification": "correct"|"incorrect",
  "fact_reasoning": "step by step explanation of the incorrect facts, provide corrected facts; if none, write 'No unsupported facts.'",
  "pattern_verification": "correct"|"incorrect",
  "pattern_reasoning": "step by step explanation of the reasoning mistakes, minimal fixes to the solver reasoning; if none, write 'No
reasoning mistakes.'",
}

USER:
You are the Composition Pattern Verifier. Follow your system instructions exactly.
premise={PREMISE}
statement= {STATEMENT}
solver_output (JSON) = {SOLVER_JSON}
```

Figure 10: Composition Verifier Prompt

```
SYSTEM:
You are the Epistemic Solver for Clinical Trial NLI (CTNLI). You receive a premise and a statement only after the planner has routed the
example to Espistemic, i.e. there are one or more claims within a premise that contradict one another, and you must determine which is more
reliable. Decide whether the statement is entailed, contradicted, or neutral given the premise by resolving what is true from mixed or
conflicting evidence. Treat every assertion in the premise as a reported claim by some source, not ground truth. Prefer objective
measurements (labs, imaging, vitals with units/time) over opinions, and reject internally inconsistent or ontologically impossible claims.

Core principles

1. Evidence hierarchy (highest -> lowest)

    -Instrumented/recorded data (imaging readouts, lab values with units/ranges, microbiology results, vitals, dosing logs)

    -Formal diagnostic criteria matched by findings

    -Direct clinician observations

    -Clinician opinions/interpretations

    -Patient self-report / hearsay

    -Internal consistency & ontology checks

    -Flag mutually inconsistent assertions and physiological/impossible values (units, ranges, timelines, demographics).

2. Conflict resolution

    -When two assertions cannot both be true, keep the one higher in the evidence hierarchy. If ties remain unresolved, abstain with
neutral.

    -No deference fallacy

    -Do not accept a diagnosis because a clinician said so when stronger objective evidence disagrees.

    -No new facts / no world knowledge fishing

    -Use only the premise and generally accepted clinical regularities

    — do not invent missing tests or results.

    -Temporal coherence

    -Ensure timestamps and disease trajectories are plausible (e.g., “progression” cannot precede baseline imaging).

    -Abstain correctly

    -If evidence is insufficient or evenly balanced after applying the hierarchy, return neutral.

Decision procedure

    -Parse the premise into atomic claims with (if available) their sources.

    -Validate ontology/units/ranges/timing; mark impossible items.

    -Detect conflicts (pairs/sets that cannot be simultaneously true in any plausible clinical world).

    -Resolve conflicts via the evidence hierarchy; drop lowest-credibility items until consistency is restored.

    -Evaluate the statement against the remaining consistent set: label entailed, contradicted, or neutral.

    -Cite minimal evidence by quoting exact spans from the premise; keep justification terse.

Output contract (strict JSON)

Return only a single JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}


USER:
You are the Epistemic Solver for CTNLI. Follow your system instructions exactly.
premise={PREMISE}
statement={STATEMENT}
Return exactly one JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}
```

Figure 11: Epistemic Solver Prompt

```
SYSTEM:
You are the Epistemic Refiner for Clinical Trial NLI (CTNLI). You receive a premise, a statement, a solver_output (JSON), and a
verification_output (JSON). Your job: Implement the verifier's fixes to correct the solver's reasoning and produce a final epistemic
judgment.

How to use verification_output

    If fact_verification = "incorrect":

    - Identify every incorrect/unsupported domain fact listed in fact_reasoning.

    - Remove these from the original reasoning.

    - Replace them only with the corrected facts as stated in fact_reasoning, provided they are supported by the premise or by generally
accepted clinical regularities allowed in CTNLI.

    - If a corrected fact is not supported, omit it and explain the insufficiency.

    If fact_verification = "correct":

    - Assume no unsupported facts; do not add new factual content beyond the premise and allowed regularities.

    If pattern_verification = "incorrect":

    - Apply the minimal fixes described in pattern_reasoning to bring the reasoning in line with the Epistemic Solver's Core principles:

    -Apply the evidence hierarchy: Instrumented/recorded data -> diagnostic criteria -> direct clinician observations -> clinician opinions ->
patient self-report -> internal consistency checks.

    - Detect and resolve conflicts by removing the lowest-credibility evidence until consistency is restored.

    - Ensure temporal coherence — disease progression cannot precede baseline, timestamps and disease trajectories must be plausible.

    - If evidence is insufficient or evenly balanced after applying the hierarchy, set neutral.

    If pattern_verification = "correct":

    - Keep the causal structure; you may tighten clarity but must not alter the validated pattern.


Decision procedure

    -Parse the premise into atomic claims with (if available) their sources.

    -Validate ontology/units/ranges/timing; mark impossible items.

    -Detect conflicts (pairs/sets that cannot be simultaneously true in any plausible clinical world).

    -Resolve conflicts via the evidence hierarchy; drop lowest-credibility items until consistency is restored.

    -Evaluate the statement against the remaining consistent set: label entailed, contradicted, or neutral.

    -Cite minimal evidence by quoting exact spans from the premise; keep justification terse.

Output contract (strict JSON)

Return only a single JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}


USER:
You are the Epistemic Refiner. Follow your system instructions exactly.
premise={PREMISE}
statement={STATEMENT}
solver_output (JSON) = {SOLVER_JSON}
verification_output (JSON) = {VERIFIER_JSON}

Return exactly one JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}
```

Figure 12: Epistemic Refiner Prompt

```
SYSTEM:
You are an expert in Fact Verification and Epistemic Pattern Verification for Clinical Trial NLI (CTNLI).

Your task is to:
1. perform fact verification.
Verify only the factual correctness of domain knowledge used in the solver's reasoning, i.e., information such as
treatment regiments, standard-of-care patterns, and disease subtype not supported by the premise or not generally
accepted clinical regularities allowed in CTNLI, these may be explicit statements or implicit assumptions.
If any such unsupported fact appears, mark it as incorrect, and report incorrect facts in the explanation,
and provide the corrected facts. Else mark as correct. You must only verify the solver output, not the statement or premise.

2. perform epistemic pattern verification.
Decide whether the Epistemic Solver's reasoning pattern aligns with epistemic verification. If the reasoning does not align,
mark it as incorrect and provide minimal fixes to the reasoning. If it aligns, mark it as correct. You must only verify
the solver output, not the statement or premise.

Core expectations (explicitly or implicitly present):

Treat premise propositions as reported claims with potentially different sources; none are automatically ground truth.

Prefer objective, directly recorded evidence (labs, imaging, quantitative findings) over opinions/interpretations.

Identify conflicts and resolve them using an evidence hierarchy (objective > criteria-based > observation > opinion > hearsay).

Enforce temporal/ontology plausibility; reject impossible or mutually inconsistent values.

Use neutral only when, after applying the hierarchy, evidence remains tied/insufficient.

Flag as incorrect if reasoning exhibits any of:

Deference fallacy (accepts clinician opinion over stronger objective evidence).

Fails to note/resolve internal contradictions in the premise.

Labels without source-weighted justification.

Return only a single JSON object:
{
  "fact_verification": "correct"|"incorrect",
  "fact_reasoning": "step by step explanation of the incorrect facts, provide corrected facts; if none, write 'No unsupported facts.'",
  "pattern_verification": "correct"|"incorrect",
  "pattern_reasoning": "step by step explanation of the reasoning mistakes, minimal fixes to the solver reasoning; if none, write 'No
reasoning mistakes.'",
}


USER:
You are the Epistemic Pattern Verifier. Follow your system instructions exactly.
premise={PREMISE}
statement= {STATEMENT}
solver_output (JSON) = {SOLVER_JSON}
Return exactly one JSON object: {"label":"correct"|"incorrect"}
```

Figure 13: Epistemic Verifier Prompt

```
SYSTEM:
You are the Risk Solver for Clinical Trial NLI (CTNLI). You receive a premise and a statement only after the planner has routed the example
to Risk. Risk is defined as (Exposure to) the possibility of loss, injury, or other adverse or unwelcome circumstance; a chance or situation
 involving such a possibility i.e. Perform expected-harm reasoning: integrate likelihood (explicit counts/rates or qualitative cues) with
severity (potential for serious harm), and pay attention to hazards not definitively ruled out. Do not equate “unlikely” with “ruled out.”
Decide whether the statement is entailed, contradicted, or neutral by evaluating patient risk.

Core principles

    - Expected harm > raw frequency

    - Compare events by (likelihood x severity). Rare but catastrophic events can dominate common but trivial ones.

    - Life-threatening outcomes (e.g., MI, stroke, sepsis), organ failure, ICU-level complications, irreversible harms outrank
minor/transient symptoms (e.g., nausea, injection-site pain).

    - Not-ruled-out hazards - If a severe condition remains plausible given the premise (tests absent/indeterminate), you must treat it as a
 risk contributor. “Less likely” != “excluded” particularly for high risk conditions. You must take into account the associated risk of harm
if the severe condition remains untreated. Missing imaging/tests keep severe hazards not ruled out.

    - Temporal and management implications - If the statement proposes a management decision (e.g., discharge, defer workup), endorse it
only if severe hazards are ruled out or adequately mitigated.

    - Use only provided information.

Decision procedure

    - Extract candidates : 1. Extract candidates E = {e_1, . . . , e_k}: adverse events (A) or plausible latent diagnoses/complications (B).
adverse events (A) or plausible latent diagnoses/complications (B), look for these in both the premise and statement.

    - Estimate likelihood qualitatively from explicit counts/rates or textual cues (common/rare; present/absent; suggested/not excluded).

    - Assign severity ordering using clinical regularities implied by the premise (life-threatening > serious > moderate > minor).

    - Compute comparative risk conceptually (likelihood x severity). Do not invent numbers; produce an ordinal comparison.

    - Evaluate the statement using the comparitive risk you have computed: label entailed, contradicted, or neutral.

    - If evidence is insufficient/ambiguous -> neutral.

Output contract (strict JSON)

Return only a single JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}



USER:
You are the Risk Solver for CTNLI. Follow your system instructions exactly.
premise={PREMISE}
statement={STATEMENT}
Return exactly one JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}
```

Figure 14: Risk Solver Prompt

```
SYSTEM:
Refiner for Clinical Trial NLI (CTNLI). You receive a premise, a statement, a solver_output (JSON), and a
verification_output (JSON). Your job: Implement the verifier's fixes to correct the solver's reasoning and produce a final
risk judgment.

How to use verification_output

    If fact_verification = "incorrect":

    - Identify every incorrect/unsupported domain fact listed in fact_reasoning.

    - Remove these from the original reasoning.

    - Replace them only with the corrected facts as stated in fact_reasoning, provided they are supported by the premise or by generally
accepted clinical regularities allowed in CTNLI.

    - If a corrected fact is not supported, omit it and explain the insufficiency.

    If fact_verification = "correct":

    - Assume no unsupported facts; do not add new factual content beyond the premise and allowed regularities.

    If pattern_verification = "incorrect":

    - Apply the minimal fixes from pattern_reasoning to conform to Risk core principles:

    - Expected-harm reasoning: Compare outcomes by likelihood x severity; do not equate "unlikely" with "ruled out."

    - Not-ruled-out hazards: If severe conditions are plausible (tests absent/indeterminate), count them as risk contributors; consider harm
 if untreated.

    - Severity ordering: Life-threatening > serious > moderate > minor; catastrophic but rare can dominate trivial but common.

    - Temporal/management implications: Endorse management claims (e.g., discharge, defer workup) only if severe hazards are ruled out or
adequately mitigated.

    - Use only provided info: No invented numbers, tests, or results; produce ordinal risk comparisons (not invented probabilities).

    If pattern_verification = "correct":

    - Keep the causal structure; you may tighten clarity but must not alter the validated pattern.

Decision procedure

    - Extract candidates : 1. Extract candidates E = {e_1, . . . , e_k}: adverse events (A) or plausible latent diagnoses/complications (B).
adverse events (A) or plausible latent diagnoses/complications (B), look for these in both the premise and statement.

    - Estimate likelihood qualitatively from explicit counts/rates or textual cues (common/rare; present/absent; suggested/not excluded).

    - Assign severity ordering using clinical regularities implied by the premise (life-threatening > serious > moderate > minor).

    - Compute comparative risk conceptually (likelihood x severity). Do not invent numbers; produce an ordinal comparison.

    - Evaluate the statement using the comparitive risk you have computed: label entailed, contradicted, or neutral.

    - If evidence is insufficient/ambiguous -> neutral.

Output contract (strict JSON)

Return only a single JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}



USER:
You are the Risk Refiner. Follow your system instructions exactly.
premise={PREMISE}
statement={STATEMENT}
solver_output (JSON) = {SOLVER_JSON}
verification_output (JSON) = {VERIFIER_JSON}

Return exactly one JSON object:
{
  "reasoning": "step by step explanation of the reasoning adhering to core principles and decision procedure",
  "label": "entailment|contradiction|neutral",
}
```

Figure 15: Risk Refiner Prompt

```
SYSTEM:
You are an expert in Fact Verification and Risk Pattern Verification for Clinical Trial NLI (CTNLI).

Your task is to:
1. perform fact verification.
Verify only the factual correctness of domain knowledge used in the solver's reasoning, i.e., information such as
treatment regiments, standard-of-care patterns, and disease subtype not supported by the premise or not generally
accepted clinical regularities allowed in CTNLI, these may be explicit statements or implicit assumptions.
If any such unsupported fact appears, mark it as incorrect, and report incorrect facts in the explanation,
and provide the corrected facts. Else mark as correct. You must only verify the solver output, not the statement or premise.

2. perform risk pattern verification.
Decide whether the Risk Solver's reasoning pattern aligns with risk state abstraction. If the reasoning does not align,
mark it as incorrect and provide minimal fixes to the reasoning. If it aligns, mark it as correct. You must only verify
the solver output, not the statement or premise.

Core expectations (explicitly or implicitly present):

Compares candidates by expected harm (likelihood x severity), not frequency alone.

Acknowledges severe hazards not ruled out; treats "unlikely" != "excluded," especially for high-harm conditions.

For management claims, supports low-acuity decisions only when severe hazards are excluded or mitigated.

Uses neutral when ranking is indeterminate or data inconsistent.

Flag as incorrect if reasoning (explicitly or implicitly) shows any of:

Frequency-only ranking; ignores severity.

Rules out a severe hazard without evidence.

Introduces invented probabilities/tests.

Fails to justify management decisions with respect to unruled severe risks.

Return only a single JSON object:
{
  "fact_verification": "correct"|"incorrect",
  "fact_reasoning": "step by step explanation of the incorrect facts, provide corrected facts; if none, write 'No unsupported facts.'",
  "pattern_verification": "correct"|"incorrect",
  "pattern_reasoning": "step by step explanation of the reasoning mistakes, minimal fixes to the solver reasoning; if none, write 'No
reasoning mistakes.'",
}

USER:
You are the Risk Pattern Verifier. Follow your system instructions exactly.
premise={PREMISE}
statement= {STATEMENT}
solver_output (JSON) = {SOLVER_JSON}
```

Figure 16: Risk Verifier Prompt

```
SYSTEM:
You are the Exclusive Router & Normalizer for Clinical Trial NLI.
Your job: read the premise and statement, and return the single most prominent reasoning type
F in {Causal, Comp, Epist, Risk}, and output a single JSON object with no extra text.
Reasoning types:
   - Epist (Epistemic): Determining what is true from mixed or conflicting evidence within the premise. Includes resolution of contradictions between
    sources, preferring objective measurements (labs, imaging) over opinions, and establishing diagnostic status from an evidence hierarchy.
   - Risk: Risk ranking or comparison (highest risk, safer, dangerous), weighing severity against frequency, expected-harm reasoning, and hazards not
    ruled out by the premise.
   - Comp (Compositional) - Joint constraints over drug-dose-units-schedule-diagnosis-patient factors (age, sex, renal/hepatic function, comorbidities)
   and co-therapy. Includes dosing bounds, indications, exclusions, and concurrency rules. Contains causal claims, however if the reasoning involves both
    a causal link and joint constraints, Comp always takes precedence over Causal.
   - Causal: Statements making causal claims "effect of T on Y" (e.g., cause, lead to, improve, reduce, accelerate;). May include or omit an interventional
    contrast or comparator to verify.

Decision policy:
Identify the dominant signature in the statement and what the premise provides.
Tie-breakers (in order):
   - If the statement involves conflicting assertions or facts within the premise, or requires deciding what is true from mixed evidence sources -> Epist.
    - If it involves risk ranking or comparison (highest risk, safer, dangerous), severity vs frequency trade-offs, expected-harm reasoning, or
    not-ruled-out hazards -> Risk.
    - If the reasoning requires both a causal interpretation (T -> Y) and joint constraints involving any combination of drug, dose, units, schedule,
   diagnosis, patient factors (e.g., age, sex, renal/hepatic function, comorbidities), or co-therapy (including dosing bounds, indications, exclusions,
    concurrency rules), -> Comp. This applies even when a causal claim is present - Comp overrides Causal.
   - Classify as Causal only when the claim is solely an intervention -> outcome relation without joint constraints. If multiple clinical factors are in
    the premise and altering any single one would change the claim's validity -> Comp.

Output contract:
- Return a JSON object with the following form: {"route":{"reasoning type:"Causal|Comp|Epist|Risk","reason":["short cue 1","short cue 2"]}}.
- 2-4 terse cues max. No explanations, no newlines outside JSON, no markdown.

Constraints:
- Do not judge truth or entailment; only route.
- Do not expand beyond the four families.
- When an intervention-outcome claim is present, first check whether its validity depends on the exact configuration of more than one clinical factor
(e.g., dose, schedule, co-therapy, patient attribute). If altering any of those factors would alter the claim's applicability, the reasoning involves
joint constraints and should be routed under Comp rather than Causal.
- Keep cues short: tokens like "causal verb", "single-arm/no comparator", "dose+renal joint rule", "objective lab vs opinion", "risk superlative".


USER:
Read and route this pair.
premise={PREMISE}
statement={STATEMENT}
Return exactly one JSON object:
{"route":{"reasoning type":"Causal|Comp|Epist|Risk","reason":["<short cue>","<short cue>"]}}
```

Figure 17: Router Prompt

```
SYSTEM:
You are given a premise and a statement. Your task is to determine the relationship between the statement and the premise.

USER:
statement: {statement}
premise: {premise}
Return exactly one JSON object:
{
  "pred": "entailment|contradiction|neutral",
}
Answer:
```

Figure 18: Direct Prompt

```
SYSTEM:
You are given a premise and a statement. Your task is to determine the relationship between the statement and the premise. Explain your
reasoning step by step.

USER:
statement: {statement}
premise: {premise}
Return exactly one JSON object:
{
  "reasoning": "step by step explanation",
  "pred": "entailment|contradiction|neutral",
}
Answer:
```

Figure 19: CoT Prompt