CLINICAL HALLUCINATION DETECTION WITH COVERAGE-GATED ONTOLOGY CHECKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) that write clinical text can state false facts in a fluent, convincing form. A common safeguard is a second LLM, an auditor, that checks each claim by calling external knowledge tools. In practice this is fragile: auditors misuse the tools or misread their replies, and some models, including the medical model we test, cannot call tools at all. We propose coverage-gated ontology checking, which takes the knowledge lookup out of the auditor's hands. Code first decides whether a clinical ontology can speak to a claim. If it can, the ontology checks the claim, and the auditor reads the results and gives a single verdict, without ever calling a tool. We evaluate the approach on fourteen auditors from seven model families, from 3B to 128B parameters. On a clinical hallucination benchmark it improves every auditor, raising the Matthews correlation coefficient by 0.09 to 0.30 over the same auditor without checks. The gains are largest for weaker models, and with checks a 7B auditor matches 70B and 120B auditors without them. The approach also works for models that cannot call tools, and it outperforms tool calling for most models that can. Because coverage is decided before the auditor runs, it shows in advance where the ontology can help.

1 INTRODUCTION

A language model writing clinical text can state something false in a clinically fluent form: a drug indicated for a condition it is contraindicated in, a laboratory value outside the range compatible with life, a causal relation with its direction reversed. A common response is to audit the output with a second language model and to give that auditor external knowledge as callable tools.

That arrangement is usually evaluated on a single auditor, leaving the questions that decide whether it is worth deploying unanswered. Does the knowledge help every auditor or only weak ones? Does the benefit survive as models improve? Is it the structured knowledge that helps, the terminology it is built from, or the prompt that accompanies it? And to which claims does it apply at all?

We answer them with a simpler arrangement that takes the knowledge lookup out of the auditor's hands. Tool calling is fragile in practice: auditors issue malformed calls or misread replies that are often "unknown", and some models, including the medically tuned one we test, expose no tool-calling interface at all. In our approach, code first decides from the claim's text whether the ontology can speak to it, runs the applicable checks, and passes their results to the auditor, which returns a single verdict without calling a tool (Figure 1). Because that coverage decision is made before the auditor runs, it also shows in advance which claims the checks can help with.

We evaluate it on a clinical hallucination benchmark, using an OWL 2 ontology whose checks decide drug–condition contraindication, physiologic plausibility of laboratory values, and relation direction. Our contributions:

- **Coverage-gated ontology checking.** A verification pass that takes the knowledge lookup out of the auditor's hands: code decides from the claim's text whether the ontology can speak to it, runs the applicable checks, and gives the auditor their results for a single verdict. The auditor never calls a tool (Section 2.1).
- **Gains across fourteen auditors.** On a clinical hallucination benchmark the method improves every auditor tested, from seven model families and 3B to 128B parameters, by 0.091 to 0.303

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

MCC over the same auditor without checks. Weaker auditors gain most, and with checks a 7B auditor matches 70B and 120B auditors without them (Section 4.1).

- **Verification without tool calling.** The method works for auditors that cannot call tools, including the medically tuned model, and outperforms tool calling on nine of the eleven auditors that can (Section 4.2).
- **Knowing in advance where checks apply.** Coverage is decided before the auditor runs, from the text alone, so it identifies ahead of time the claims the ontology can help with (Section 4.4).

2 Ontology-Grounded Checking

2.1 The ontology and its checks

The knowledge base is an OWL 2 DL ontology (W3C OWL Working Group, 2012) of 38 classes and 4 object properties, built with `owlready2` (Lamy, 2017) and verified consistent by the HermiT reasoner (Glimm et al., 2014). Of its four layers, the experiments exercise two: an upper taxonomy of hallucination types, and domain facets for pharmacology, diagnostics and laboratory medicine, each anchored to an external terminology by `skos:exactMatch`. Harm tiers and generation-mechanism classes are defined but unused here.

Three assertion families are decidable against it, each backed by a distinct source whose provenance bears on how the results should be read. **Forbidden relations** are backed by 616 (drug, contraindicated-condition) pairs over 165 drugs, mined from the MED-RT terminology through the UMLS REST API using the relations `ci_with`, `contraindicated_in`, `contraindicated_with_disease` and `may_be_contraindicated_for`; a claim that a drug treats a condition it is contraindicated in is a violation. For 20 common drugs a curated indication list is also consulted, and a treatment claim outside it is flagged as unsupported. **Threshold violations** use physiologic plausibility bounds for 15 common laboratory analytes, from clinical pathology references and LOINC. **Inverted relations** are causal or therapeutic relations asserted in the wrong direction, flagged when the reverse relation appears in the indication list or a curated table of 31 causal relations and the forward one does not.

From text to assertions. A lexicon-backed extractor parses a claim into typed entities, binary relations with a canonical predicate, and numeric assertions of the form (analyte, value, unit); an entity that does not resolve yields no assertion. Each assertion is checked against the sources above, and each one flagged becomes an individual of the corresponding hallucination class in an ABox, which HermiT classifies against the TBox to infer its superclasses and harm tiers, so a violation reaches the auditor with a type rather than only a flag. The decision itself is the lookup; the reasoner supplies the typing. Section 4.2 returns to what this means for interpreting the controls.

What the checks return. Each returns a decision or `unknown`, and `unknown` is frequent and correct: it is the honest answer whenever a claim falls outside the three families, as most clinical prose does. An auditor handed a raw `unknown` must separate "this claim is sound" from "this ontology cannot speak to this claim", which the reply does not say. The gate removes that failure.

Coverage, decided from the text. A claim is *covered* when it asserts something inside the fragment the ontology axiomatizes. Concretely, coverage holds when any of three conditions is met: the extractor produces a relation whose predicate the ontology defines and whose subject and object both resolve in the lexicon; it produces a numeric assertion over an analyte the ontology has bounds for; or the translator builds at least one ABox individual. Coverage is computed from the whole text submitted for audit, so for grounded MedHallu it includes the supporting passage, which is why the same claims are covered more often with their passages than without (Table 1).

Coverage does not depend on which way any check went, which is a design constraint rather than an implementation detail. The tempting alternative, treating a claim as covered once a check has found a violation, is circular: it consults the answer to decide whether to look, and opens the gate only on claims that turn out false. The third condition is the one place a check result enters, since individuals are built for flagged assertions, but every flagged forbidden, inverted or threshold assertion already meets one of the first two; the third adds only claims with a negated clinical entity, a candidate modifier omission whatever the claim's truth (one claim on our benchmark, 13 to 22 on each external one). Table 1 reports open rates by label to show the gate does not track it.

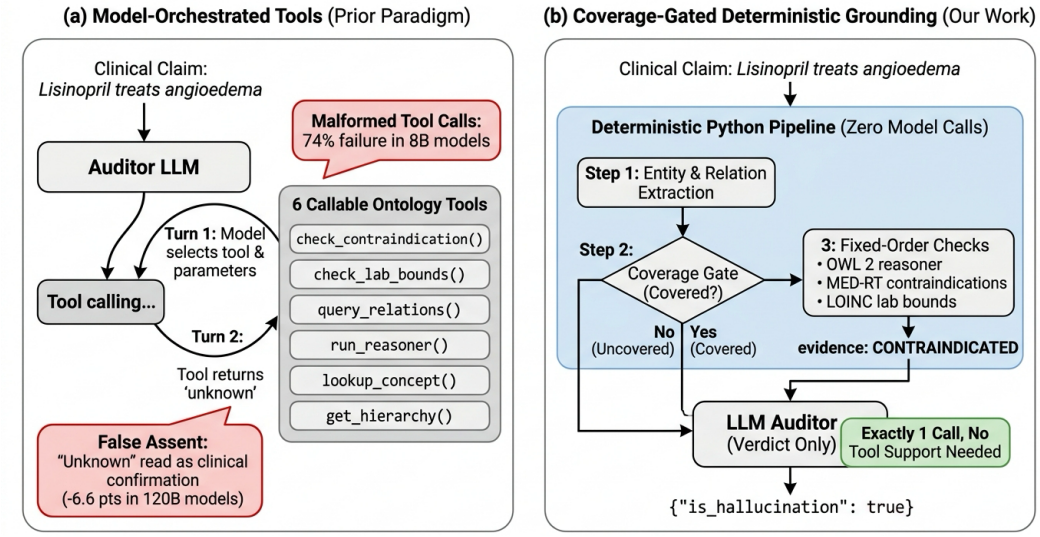


Figure 1: Two arrangements on one claim. **(a)** Tool calling, as in tool-augmented LLMs (Schick et al., 2023; Qin et al., 2024): the auditor picks tool calls and reads their often "unknown" replies; both steps fail in practice (grounded MedHallu: llama3.1:8b lost 299 of 400 verdicts, gpt-oss:120b 6.6 accuracy points; Appendix B). **(b)** Ours: coverage is settled from the audited text, checks run in code in fixed order, and the auditor, called once with the results, needs no tool-calling interface.

Table 1: Coverage gate open rates, computed from the audited text (for grounded MedHallu, including its passage) with no model involved. Rates on hallucinated and true claims differ by at most 8.3 points, in both directions, and never significantly (Fisher's exact test, $p \geq 0.11$).

Benchmark	n	Gate open	open | hallucinated	open | true
CHO gold set	365	87.1%	88.3%	80.0%
MedHallu, grounded	400	9.8%	8.5%	11.0%
MedHallu, ungrounded	400	3.8%	3.0%	4.5%
Med-HALT FCT	400	7.2%	7.5%	7.0%

The verification pass. The pass is fixed, with exactly one model call (Figure 1): the claim is parsed, coverage computed, and if the gate is open the applicable checks run in a fixed order, reasoner first, then relations, then numerics, at most three of each. Decided results become short evidence lines appended to the prompt; undecided ones are discarded. The auditor is then called once for a single JSON object carrying a boolean verdict, an error class, a confidence level and a one-sentence rationale. Where the serving stack supports constrained decoding we request a JSON object; where it does not, the prompt alone specifies the format, and scoring is identical either way.

Three consequences follow, each tested in Section 4: the auditor cannot fail by emitting a malformed call, never receives an `unknown` reply, and the pass runs unchanged on models whose serving stacks expose no function-calling interface. One form of silence still reaches it: a claim passing every check is reported as clean, which misleads weak auditors when the table lacks the fact (Section 4.3).

3 Experimental Setup

3.1 Design, claim sets and auditors

Design. The design is a within-auditor comparison of five verification arms over a fixed claim set. Each auditor sees every claim under every arm, so arm effects are estimated within a model and are not confounded by differences between models. Arms are interleaved per claim rather than run in blocks, so that a serving interruption degrades all arms equally rather than truncating one. Decoding is deterministic (temperature 0) throughout, so repeated runs of an arm reproduce its verdicts and the only variation between arms is the intended one.

Claim sets. The primary claim set contains 365 clinical assertions, 315 labelled hallucinated across five error classes and 50 labelled correct. The hallucinated claims are *synthetic*: each was generated by taking a fact from one of the ontology's backing tables and negating or perturbing it, so that a forbidden-relation claim states a drug treats a condition the terminology records as contraindicated for it. The class distribution is uneven by construction, with forbidden relations dominating (200 claims), followed by threshold violations (54), inverted relations (36), fabricated entities (24) and a single modifier omission. Section 4.3 measures the consequence of this construction directly, and Section 3.2 reports what it means for interpretation.

Three further claim sets, none built by us nor derived from the ontology, test whether findings survive off its domain. MedHallu pairs biomedical questions with answers and the passages they should be checked against (Pandit et al., 2025); we evaluate it with those passages present and removed, since dropping them turns verification into recall. Med-HALT's factual-conflict subset supplies examination items (Pal et al., 2023). Each is balanced at 400 claims by pairing every incorrect answer with a correct one.

Auditors. Fourteen models span seven families, Llama, Qwen, Gemma, Granite, Nemotron, gpt-oss and Mistral, and 3B to 128B parameters (Table 2). Sizes are total parameters; a mixture-of-experts auditor activates far fewer per token, gpt-oss:120b about 5B (OpenAI, 2025). They are served through two stacks: a local Ollama deployment at 4-bit quantization, and a separate OpenAI-compatible server hosting the 122B model. Capabilities are read from each serving stack rather than assumed, which matters because they vary in ways that bear on the method: two auditors expose no function-calling interface at all, so the tool-calling arm cannot be run for them, and seven emit explicit reasoning before answering. Reasoning auditors required a larger completion budget: at the default 4096 tokens several were truncated mid-reasoning and lost the verdict, which scores as an error and would have been misread as the method failing. The budget was raised to 16,384 tokens, and to 49,152 for the 122B, and each run records its budget. Truncated verdicts were retried with it.

3.2 Arms, measures and threats to validity

Arms. Five arms isolate the contributions of the prompt, the knowledge, and the protocol below. Two are baselines: `no_tools` is the auditor's parametric memory under the original prompt, and `cho_tools` follows the standard tool-calling paradigm (Schick et al., 2023; Qin et al., 2024), exposing the ontology as six callable tools. Three are controls introduced because the first results were uninterpretable without them. `no_tools_json` holds the verdict-only prompt fixed and removes all knowledge, so any effect of rewording is attributed to the wording; `generic_context` replaces the ontology with plain concept lookup over the same lexicon, holding prompt, protocol and gate fixed, to separate recognizing terms from deciding claims; and `cho_context` is the method.

Measures. We report accuracy, balanced accuracy and the Matthews correlation coefficient. The primary claim set is positive-heavy at roughly six to one, so MCC, not accuracy, carries the comparisons; balanced accuracy, reported alongside, responds differently to the 50 controls.

Which claims are scored. Three rules produce every table. A failure of the serving stack rather than the model (a dropped connection, a rejected request, including a model without tool support, or a verdict cut off at the completion budget) is retried, and a claim that still fails is excluded from that arm. Any other missing or unparsable verdict, including a malformed tool call, is scored as an error rather than discarded, the conservative choice: discarding would reward an arm for abstaining where it is least able to answer. Within an auditor, arms are compared only on claims every compared arm scored, so differences are attributable to the arm. Table 2 gives the resulting n: 362 to 365 for twelve auditors, 281 for qwen3.5:122b, whose other claims exceeded even the raised budget, and 254 for mistral-medium-3.5, whose run is incomplete. Other tables state their n where it differs.

The headline quantity is the *gain*, defined as the MCC of `cho_context` minus that of `no_tools_json`: both share a prompt and a protocol and differ only in whether the ontology is consulted, so the difference isolates the knowledge.

Uncertainty is reported at two levels. For a single auditor, intervals come from a paired bootstrap over claims with 2,000 replicates, recomputing both arms on each draw, which preserves the pairing an unpaired interval would discard and assumes nothing about the sampling distribution of a correlation coefficient. For the claim that every auditor improves, the unit is the auditor, and we use an

Table 2: CHO gold set, MCC, on the n claims all of an auditor's arms completed. Gain: `cho_context` minus the prompt-matched `no_tools_json`; every bootstrap interval excludes zero. vs orig.: the same gain against `no_tools`; † interval includes zero. Cap.: T tool calling, R reasoning. Dashes: arms an auditor cannot run (no tool support) or that were not run (llama3.1:8b).

Auditor	Cap.	n	`no_tools`	`+json`	`cho_tools`	`cho_context`	Gain	vs orig.
llama3:8b	—	365	0.590	0.468	—	0.772	**+0.303**	+0.181
qwen3.8:27b	T+R	365	0.707	0.676	0.811	0.907	**+0.231**	+0.200
qwen2.5:7b	T	365	0.580	0.586	0.638	0.815	**+0.228**	+0.234
medgemma:27b	—	365	0.676	0.653	—	0.877	**+0.224**	+0.200
gemma4:31b	T+R	364	0.740	0.734	0.825	0.926	**+0.192**	+0.186
qwen3.5:122b	T+R	281	0.725	0.763	0.849	0.949	**+0.187**	+0.224
llama3.1:70b	T	365	0.799	0.680	0.799	0.864	**+0.184**	+0.065†
qwen3.5:35b	T+R	362	0.764	0.758	0.911	0.916	**+0.159**	+0.153
granite4.1:3b	T	365	0.524	0.613	0.777	0.761	**+0.148**	+0.238
granite4.1:30b	T	365	0.817	0.786	0.921	0.932	**+0.146**	+0.115
nemotron3:33b	T+R	365	0.797	0.731	0.879	0.874	**+0.143**	+0.077†
gpt-oss:120b	T+R	365	0.784	0.805	0.914	0.944	**+0.139**	+0.160
mistral-medium-3.5	T+R	254	0.835	0.846	0.926	0.954	**+0.108**	+0.119
llama3.1:8b	T	365	—	0.733	—	0.825	**+0.091**	—

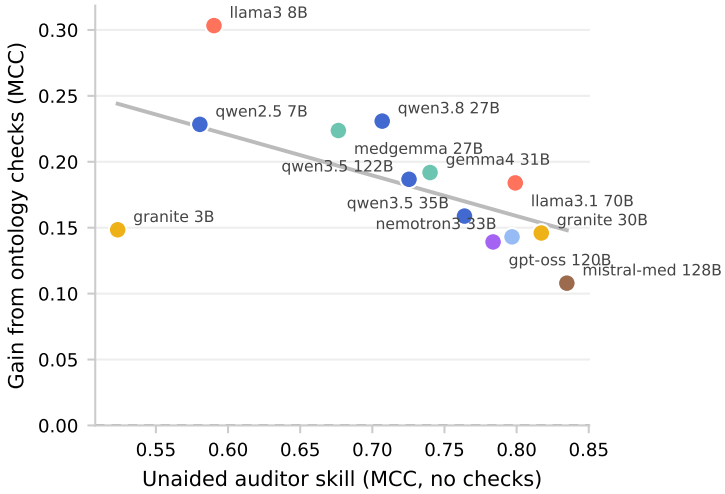


Figure 2: Gain from ontology checks (over the prompt-matched control) against unaided auditor skill, one point per auditor, coloured by model family; llama3.1:8b, with no unaided run, is not plotted. Every point lies above zero. The trend is downward, so the auditors that need the help most receive the most.

exact one-sided sign test, chosen over a t-test because gains from 3B to 128B models are not draws from a common distribution, and over Wilcoxon because it assumes no symmetry.

Threats to validity. Three threats bear on interpretation, each addressed by a measurement, not an argument. *Construct validity*: the primary claim set was synthesized from the tables the checks consult, so an apparent ability to decide claims may be retrieval of their generating facts; Section 4.3 deletes those facts and re-measures. *Label validity*: contraindication relations were mined automatically, discarding the context that qualifies them, so a relation recorded as a contraindication in one setting may be standard therapy in another; Section 4.3 reports an annotator-reviewed error rate. *External validity* is bounded by coverage, so we report three independently built claim sets.

4 Results

4.1 Checking helps every auditor

Table 2 reports MCC by arm for each auditor; the gain column is `cho_context` minus `no_tools_json`, which holds the prompt fixed, so it measures the ontology rather than the word-

Table 3: Controls, MCC on the CHO gold set, on the claims all three arms completed (279 for qwen3.5:122b, 365 otherwise). Concept lookup over the same vocabulary moves MCC by -0.017 to $+0.036$ over the prompt-only control; checks that decide the claim add 0.117 to 0.231 over lookup.

Auditor	`no_tools_json`	`generic_context`	`cho_context`	Checks − lookup
qwen3.8:27b	0.676	0.676	0.907	**+0.231**
medgemma:27b	0.653	0.653	0.877	**+0.224**
qwen2.5:7b	0.586	0.622	0.815	**+0.193**
qwen3.5:122b	0.762	0.762	0.949	**+0.187**
granite4.1:3b	0.613	0.596	0.761	**+0.165**
gpt-oss:120b	0.805	0.827	0.944	**+0.117**

ing that accompanies it, and Figure 2 plots it against unaided skill. **The gain is universal but uneven:** it is positive for all fourteen auditors, every one of the fourteen bootstrap intervals excludes zero, the narrowest margin being llama3.1:8b at $[+0.004, +0.186]$, and under a null in which an auditor is equally likely to improve or worsen the probability of fourteen of fourteen improving is $p = 6.1 \times 10^{-5}$. The prompt-matched control isolates the knowledge but is not neutral: the verdict-only prompt alone lowers llama3:8b and llama3.1:70b by about 0.12 and raises granite4.1:3b by 0.089, inflating the first two gains and deflating the third. Against the original prompt all thirteen auditors with that arm still improve, by 0.065 to 0.238, and eleven of thirteen intervals exclude zero. **Weaker auditors gain more** under either reference ($r = -0.58$ and -0.78; Pearson, $n = 13$). **Scale reduces the gain without removing it.** Within the granite family, which holds architecture and training data fixed, the 30B is far stronger unaided (0.817 against 0.524) and still gains 0.146 over the prompt-matched control and 0.115 over the original prompt, against 0.148 and 0.238 for the 3B. Whether the gain is flat across a tenfold difference in scale thus depends on the reference; that the larger model still benefits does not. **And checking compresses the differences between auditors:** unaided MCC spans 0.524 to 0.835 (population sd 0.095), with checks 0.761 to 0.954 (sd 0.063). With checks qwen2.5:7b reaches 0.815, against 0.799 for the dense llama3.1:70b and 0.784 for gpt-oss:120b (about 5B active parameters) without them. Both differences lie inside their paired bootstrap intervals, so the 7B matches rather than beats them, which for deployments that must verify locally is the result with practical consequences.

4.2 What produces the gain, and what the protocol contributes

Vocabulary or decisions. Table 3 replaces the checks with concept lookup over the same lexicon. The lookup arm shares the prompt, the protocol, the gate and the vocabulary with the checking arm, but it only tells the auditor what each resolved term is and decides nothing. Lookup moves MCC by between -0.017 and $+0.036$; the checks move it by 0.117 to 0.231 (Figure 3 in Appendix D draws this per auditor). Resolving the terminology is not what helps. Deciding the claim is.

What this control does not separate is how the decision is made. Every decision in our implementation is a table lookup (Section 2.1); HermiT types each flagged assertion and propagates its harm tiers, which the auditor receives as a class label. A flat table of the same facts would reach the same decisions, so these results support curated knowledge that decides claims, not description-logic reasoning, and we claim nothing for the reasoner beyond the typing it supplies.

The checks alone, and what the auditor adds. Scored with no model, calling a claim hallucinated exactly when a check flags it, the checks reach 0.581 MCC. They catch 192 of 200 forbidden relations and all 54 threshold violations, but 20 of 36 inverted relations and 2 of 24 fabricated entities, and they flag 6 of the 50 correct claims. Every auditor reading them scores higher, 0.761 to 0.954. A stricter rule also uses the auditor's own knowledge: the check's flag where one fires, the auditor's prompt-only verdict otherwise. Reading the checks beats that rule for thirteen of fourteen auditors, by up to 0.102, with eleven intervals excluding zero (Appendix C); granite4.1:3b is the exception at -0.006. What the auditor mostly adds is rejecting a flag on a claim that is correct.

Determinizing the protocol. Table 2 also compares the deterministic protocol with tool calling over the same checks. Deterministic checking is better for nine of the eleven auditors that support tool calling, by 0.005 to 0.177 MCC, and worse for two, granite4.1:3b by 0.016 and nemotron3:33b by 0.005. The argument for determinizing is not that tool calling is broken: a 3B model released in April 2026 runs the tool protocol cleanly and slightly outscores the deterministic pass. The argument

is that tool calling is not universally available: two auditors here cannot use it at all, and one of them is the only medically tuned model in the roster, which is precisely the kind of model a clinical deployment would reach for first.

4.3 IS THE GAIN REASONING, OR RETRIEVAL OF THE ANSWER KEY?

The benchmark's forbidden-relation claims, 200 of 365, were generated from a table of drug–condition contraindications mined from a clinical terminology, and every one of them has its generating pair in that table. The checks consult the same table. A checker that recovers the pair has not decided the claim; it has retrieved the fact that produced it, and a benchmark built this way will reward the ontology regardless of what the ontology is worth.

We therefore remove the generating pairs for a random half of those claims from the table the checks consult, holding all else fixed, and re-run. The benchmark is then split into two halves differing only in whether the knowledge base still contains the fact behind the claim. The gain on the held-out half is the part that is not retrieval.

Table 4: Held-out experiment, `cho_context`, thirteen auditors. Drops are full-table minus reduced-table accuracy on the same claims (positive is a loss); all claims are hallucinated, so accuracy is recall. Control: pair kept; treated: pair removed. Difference: treated minus control, with a bootstrap interval. Left: treated-claim accuracy under the reduced table minus the prompt-only control.

Auditor	n	Control drop	Treated drop	Difference	Left
llama3:8b	100	$+0.010$	$+0.130$ $[+0.070, +0.200]$	$+0.120$ $[+0.050, +0.190]$	-0.140
qwen3.8:27b	100	$+0.010$	$+0.060$ $[+0.020, +0.110]$	$+0.050$ $[+0.000, +0.100]$	$+0.050$
qwen2.5:7b	100	$+0.000$	$+0.020$ $[+0.000, +0.050]$	$+0.020$ $[+0.000, +0.050]$	-0.010
medgemma:27b	100	$+0.010$	$+0.120$ $[+0.060, +0.190]$	$+0.110$ $[+0.050, +0.180]$	$+0.010$
gemma4:31b	100	$+0.000$	$+0.090$ $[+0.040, +0.150]$	$+0.090$ $[+0.040, +0.150]$	$+0.010$
qwen3.5:122b	72	$+0.014$	$+0.097$ $[+0.042, +0.167]$	$+0.083$ $[+0.014, +0.153]$	$+0.046$
llama3.1:70b	100	$+0.010$	$+0.070$ $[+0.030, +0.120]$	$+0.060$ $[+0.010, +0.120]$	$+0.060$
qwen3.5:35b	98	$+0.010$	$+0.071$ $[+0.031, +0.122]$	$+0.061$ $[+0.011, +0.122]$	$+0.021$
granite4.1:3b	100	$+0.020$	$+0.190$ $[+0.120, +0.270]$	$+0.170$ $[+0.090, +0.260]$	-0.120
granite4.1:30b	100	$+0.010$	$+0.060$ $[+0.020, +0.110]$	$+0.050$ $[+0.000, +0.100]$	-0.010
nemotron3:33b	100	$+0.000$	$+0.090$ $[+0.040, +0.150]$	$+0.090$ $[+0.040, +0.150]$	$+0.000$
gpt-oss:120b	100	-0.010	$+0.040$ $[+0.000, +0.090]$	$+0.050$ $[+0.000, +0.110]$	$+0.030$
llama3.1:8b	100	$+0.050$	$+0.200$ $[+0.130, +0.280]$	$+0.150$ $[+0.060, +0.240]$	-0.190

Deleting the fact that generated a claim lowers accuracy on that claim for every auditor tested, by 0.020 to 0.170 beyond the change on claims whose facts were left alone (Table 4).

The dependency is real and general. All thirteen auditors move in the same direction (exact sign test over auditors, $p = 1.2 \times 10^{-4}$), and the control rows rule out a generic effect of shrinking the table. Across auditors, 122 verdicts on treated claims went from correct to wrong and one the other way; counted by claim, 35 treated claims lost a verdict and one gained one ($p = 5.4 \times 10^{-10}$), against 6 and 1 among the controls. If anything the control understates the difference: pairs with several surface forms split across the halves, so the deletion also removed the check result for 9 control claims. A claim set synthesized from a knowledge base credits the checker for recovering facts it was handed, so the gains in Table 2 are inflated by it. **Its size varies**, from 0.020 to 0.170, with nine of thirteen difference intervals excluding zero and four touching it.

Once the fact is gone, the checks help no one and mislead weak auditors. With the pair removed, accuracy on the treated claims is within 0.06 of the prompt-only control for ten auditors, and 0.12 to 0.19 *below* it for the three weakest, llama3:8b, llama3.1:8b and granite4.1:3b. The cause is what the checks report when the table is silent. Of the 100 treated claims, 18 are still flagged through the curated indication list and 14 through a surviving contraindication entry, both lookups, 4 carry only a negation flag, and 64 pass every check, which the reasoner reports as having found no violation. The protocol shows that to the auditor as a clean result. On those 64 claims the three weakest auditors fall from 0.948 accuracy with the prompt alone to 0.708, the other ten from 0.875 to 0.857. On this class the measured gain is retrieval, and a missing fact reaches the auditor not as silence but as assent.

This covers the forbidden-relation class, the largest at 200 of 365 claims; threshold and inverted-relation claims come from their own backing sources and plausibly carry a comparable dependency we have not separated. Threshold claims, generated from the same bounds the checks consult, show the largest per-class gain: pooled over auditors, accuracy on them rises from 0.842 to 1.000. Avoiding the problem requires drawing claims and checks from independent sources, the construction we would adopt if building this claim set again.

How accurate are the benchmark's own labels? Mining discards the context that qualifies a contraindication, so a forbidden-relation label may describe accepted therapy. Two annotators reviewed 110 claims ($\kappa = 0.966$): none of the 100 labelled hallucinated was judged correct (0/100, exact 95% upper bound 3.6%), and seven were context-dependent, true in some clinical settings and false in others. Appendix F gives the details and caveats, including that both annotators are authors.

4.4 Where the checks apply

Coverage is 87% on the benchmark the ontology was built for and 4 to 10% on three built elsewhere (Table 1). Off domain no claim is covered through a relation, only through a laboratory value or a negated entity, and the checks decide almost nothing: scored alone they reach MCC -0.054, -0.043 and $+0.046$ on ungrounded MedHallu, grounded MedHallu and Med-HALT. Nor do we find a gain there. On grounded MedHallu, the one external benchmark on which the method and its prompt-matched control were both run (llama3.1:8b, on the 288 claims every arm completed), the control leads the checking arm, 0.486 against 0.397 MCC; Appendix B lists every external run we hold. The gate stops the ontology firing where it cannot decide; it does not manufacture a benefit where there is no coverage. Uncovered claims fail for three separable reasons, each dominating a different benchmark, so a single remedy would be misdirected; Appendix A gives the breakdown.

The cost of a checked claim. We profiled energy for the tool-calling design, not the deterministic pass. On device-class hardware a claim audited with tool calls costs 150.7 J, about half the 292.6 J on an RTX 4090 and two-thirds of the 222.4 J on an H200 server GPU, which spends most of its power idling when verifying one claim at a time (Table 7 in Appendix E). The deterministic pass replaces the tool round trips with 0.02 to 0.26 s of CPU and one model call; its energy was not profiled.

5 Related Work

Hallucination and its detection. Surveys treat hallucination as a persistent property of language models rather than a defect of particular systems (Ji et al., 2023; Huang et al., 2025). Detection methods largely score a model against itself or retrieved text, decomposing generations into atomic facts (Min et al., 2023) or reading the model's uncertainty (Farquhar et al., 2024; Kossen et al., 2024). These need no curated knowledge and apply anywhere; our checks apply to a bounded set of claims and return a decision rather than a score. The two are complementary, and coverage says which regime a claim falls in.

Medical hallucination benchmarks. Med-HALT tests factual conflict and reasoning failures on medical exam material (Pal et al., 2023), and MedHallu pairs biomedical questions with supporting passages (Pandit et al., 2025). We use both as claim sets built independently of our ontology; its checks reach 4 to 10% of them, which is a boundary we report, not a result we obtain.

Knowledge-graph grounded evaluation. Benchmarks that ground hallucination evaluation in a knowledge graph are close neighbours of this work; MultiHal builds evaluation items from knowledge-graph paths (Lavrinovics et al., 2025). That construction is scalable and creates the dependency this paper measures: a checker reading the same source is not tested on its ability to decide a claim. Section 4.3 quantifies it for our claim set.

Annotation reliability in clinical data. Clinical experts labelling the same cases disagree, and the disagreement propagates: classifiers trained separately on the labels of 11 ICU consultants agreed poorly with one another on an external dataset, with an average pairwise Cohen's κ of 0.255 between the models (Sylolypavan et al., 2023). Labels mined automatically from terminology relations face a version of the same problem, since the context a clinician would supply is absent from the mined relation; Section 4.3 measures how often that matters here.

Terminologies and concept extraction. Clinical terminologies (Bodenreider, 2004; Nelson et al., 2011; Donnelly, 2006) supply vocabulary and mature tools link text to them (Neumann et al., 2019; Soldaini & Goharian, 2016; Kraljevic et al., 2021). Our lookup control is the comparison: over the same vocabulary, concept lookup moves MCC by at most 0.036 and the checks by up to 0.231. Recognizing that a token denotes a drug is not deciding whether that drug may treat that condition.

Tool-augmented models. Language models can be trained or prompted to call external tools (Schick et al., 2023; Qin et al., 2024), work that asks whether and which tool to call. We ask a prior question, whether the knowledge base can decide the claim at all, settle it before the auditor runs, and remove the choice from the model; two of our auditors cannot call tools at all.

Cascades and routing. Systems route queries between models of different cost (Chen et al., 2023; Ong et al., 2024) or exit a network early (Teerapittayanon et al., 2016), calibrated where needed by distribution-free risk control (Angelopoulos et al., 2024). These route *answering*; the compression we measure, a 7B auditor with checks matching 70B and 120B auditors without them, is what would make routing *verification* attractive. Calibrating that is future work.

6 Discussion

Where deterministic checks help in practice. Because the checks run in code, they give the same answer for the same claim every time, name the source that decided it, and cost a fraction of a second of CPU. That makes them a safeguard independent of the audited model: a flagged claim can be held back, or sent to a larger model or a clinician, with its reason. A flag is a signal rather than a verdict, which the auditor turns into one: on our benchmark the checks also flag 6 of the 50 correct claims, and auditors clear them. Because the checks never change between runs, they can also serve as a monitor: a rise in how often a model's output is flagged, or in how often an auditor overrides the checks, points to the model, a simple way to notice more hallucination or a degraded update.

Scope. Our evidence comes from a clinical benchmark synthesized from the same curated sources the checks use, spanning contraindications, laboratory ranges and relation direction; Section 4.3 measures what that construction contributes. The checks' decisions come from curated tables typed by the ontology, so the results speak for curated knowledge that decides claims, and what description-logic reasoning adds is a separate question. On independent benchmarks the checks apply to few claims, so the full method was run there with one auditor. The next step is a claim set drawn from independent sources and real clinical notes, reviewed by clinicians.

Future work. The clearest way to widen the benefit is to widen coverage. Appendix A shows what limits it off domain: vocabulary on Med-HALT, and relation extraction on MedHallu, where 70% of claims already name a known entity. Growing the ontology's vocabulary, extractor and decidable relations together would extend the checks to more clinical claims, with coverage still known before the auditor runs. This matters most for deployment on edge devices. Small auditors gain the most from the checks, and with them a 7B model already matches 70B and 120B models, so wider coverage would let a small, efficient model verify clinical text locally, keeping patient text on the device.

7 Conclusion

Giving a clinical auditor a checkable knowledge base helps every auditor we tested, from 3B to 128B and including one trained on medical text. The benefit is largest for weak auditors but does not vanish with capability: a tenfold larger model in the same family still gains, and with checks a 7B auditor matches 70B and 120B auditors without them. That matters most where verification has to run locally, which is also where the weakest auditors are.

The benefit comes from the checks and the auditor working together. Curated clinical knowledge, organized and typed by the ontology, decides the claims it covers; telling the auditor only what kind of term each word is adds almost nothing, and the checks alone fall short of every auditor that reads them. Running the checks in code rather than as tool calls lets models without tool support use the ontology, including the medically tuned model, and outperforms tool calling for most models that can call tools. Because coverage is decided before the auditor runs, the approach also shows in advance which claims it can help with.

REPRODUCIBILITY STATEMENT

Code, the ontology, the analysis scripts, and the 45,042 per-claim result rows behind every table and figure are available at `https://anonymous.4open.science/r/cho-artifact-BB8F/`. Every table and figure is regenerated from the per-claim results by a script in `evaluation/edge_cost/`, and the repository's README maps each one to its command; the figures are drawn from the result files rather than transcribed. The method is described in Section 2.1 and implemented in `context_auditor.py`; the coverage predicate is stated in full in Section 2.1 and the arms and scoring rules in Section 3.2. Data withheld under the UMLS Metathesaurus licence, namely the mined contraindication table, the UMLS-expanded lexicon, and the 200 gold claims generated from them, is regenerable with a free UTS account using the build scripts included; the README documents each withheld file and the command that rebuilds it; a hand-curated indication list for 20 drugs ships with its UMLS identifiers removed, which no check reads. Per-claim rows retain check verdicts but have UMLS-derived strings redacted for the same reason, so every reported analysis runs on the released data as shipped, except those that re-run the checks themselves (Table 1 and the checks-alone scores off domain), which need the withheld tables. No credentialed clinical data is included.

ETHICS STATEMENT

This work uses no patient data. The primary claim set is synthetic, generated by perturbing facts drawn from public terminologies, and the external benchmarks are public. Experiments on a credentialed clinical corpus were excluded from this submission and its artifact.

Two ethical considerations bear on the results themselves. First, the method is a detector for clinical errors, and a detector that is trusted beyond its coverage could do harm: we therefore measure coverage explicitly, report that gains do not transfer off it, and show in Section 4.3 that where the knowledge base is incomplete a clean check can lead a weak auditor to accept a false claim. Second, the benchmark's labels derive from automatically mined contraindications, and that mining discards clinical context. We report an annotator review of 110 claims rather than assume the labels are sound, and disclose that the annotators are authors rather than independent clinicians.

AI USE STATEMENT

Generative AI was used in this work to polish the manuscript text and code the evaluations. The authors set the research direction, conducted the experiments, reviewed and corrected the code, and conducted the clinical label review themselves, and made every judgement about what the evidence supports. We have reviewed all AI-assisted work and take responsibility for the final content of this paper, including its text, claims and artifacts.

REFERENCES

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *International Conference on Learning Representations (ICLR)*, 2024.

Olivier Bodenreider. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32(suppl_1):D267–D270, 2004. doi: 10.1093/nar/gkh061.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*, 2023.

Kevin Donnelly. SNOMED-CT: The advanced terminology and coding system for eHealth. In *Studies in Health Technology and Informatics*, volume 121, pp. 279–290. IOS Press, 2006.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, 2024. doi: 10.1038/s41586-024-07421-0.

Birte Glimm, Ian Horrocks, Boris Motik, Giorgos Stoilos, and Zhe Wang. HermiT: An OWL 2 reasoner. *Journal of Automated Reasoning*, 53(3):245–269, 2014. doi: 10.1007/s10817-014-9305-1.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, 2025. doi: 10.1145/3703155.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023. doi: 10.1145/3571730.

Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. Semantic entropy probes: Robust and cheap hallucination detection in LLMs. *arXiv preprint arXiv:2406.15927*, 2024.

Zeljko Kraljevic, Thomas Searle, Anthony Shek, Lukasz Roguski, Kawsar Noor, Daniel Bean, Aurelie Mascio, Leilei Zhu, Amos A. Folarin, Angus Roberts, Rebecca Bendayan, Mark P. Richardson, Robert Stewart, Anoop D. Shah, Wai Keong Wong, Zina Ibrahim, James T. Teo, and Richard J. B. Dobson. Multi-domain clinical natural language processing with MedCAT: The Medical Concept Annotation Toolkit. *Artificial Intelligence in Medicine*, 117:102083, 2021. doi: 10.1016/j.artmed.2021.102083.

Jean-Baptiste Lamy. Owlready: Ontology-oriented programming in Python with automatic classification and high level constructs for biomedical ontologies. *Artificial Intelligence in Medicine*, 80:11–28, 2017. doi: 10.1016/j.artmed.2017.07.002.

Ernests Lavrinovics, Russa Biswas, Katja Hose, and Johannes Bjerva. MultiHal: Multilingual dataset for knowledge-graph grounded evaluation of LLM hallucinations. *arXiv preprint arXiv:2505.14101*, 2025.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12076–12100. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.741.

Stuart J. Nelson, Kelly Zeng, John Kilbourne, Tammy Powell, and Robin Moore. Normalized names for clinical drugs: RxNorm at 6 years. *Journal of the American Medical Informatics Association*, 18(4):441–448, 2011. doi: 10.1136/amiajnl-2011-000116.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. ScispaCy: Fast and robust models for biomedical natural language processing. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pp. 319–327. Association for Computational Linguistics, 2019. URL https://aclanthology.org/W19-5034/.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs with preference data. *arXiv preprint arXiv:2406.18665*, 2024.

OpenAI. gpt-oss-120b & gpt-oss-20b model card, 2025.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Med-HALT: Medical domain hallucination test for large language models. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*, pp. 314–334, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.conll-1.21.

Shrey Pandit, Jiawei Xu, Junyuan Hong, Zhangyang Wang, Tianlong Chen, Kaidi Xu, and Ying Ding. MedHallu: A comprehensive benchmark for detecting medical hallucinations in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 2858–2873, Suzhou, China, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.143. URL https://aclanthology.org/2025.emnlp-main.143/.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, et al. ToolLLM: Facilitating large language models to master 16000+ real-world APIs. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Luca Soldaini and Nazli Goharian. QuickUMLS: A fast, unsupervised approach for medical concept extraction. In *MedIR Workshop, SIGIR*, 2016.

Aneeta Sylolypavan, Derek Sleeman, Honghan Wu, and Malcolm Sim. The impact of inconsistent human annotations on AI driven clinical decision making. *npj Digital Medicine*, 6(26), 2023. doi: 10.1038/s41746-023-00773-3.

Surat Teerapittayanon, Bradley McDanel, and H. T. Kung. BranchyNet: Fast inference via early exiting from deep neural networks. In *International Conference on Pattern Recognition (ICPR)*, pp. 2464–2469, 2016.

W3C OWL Working Group. OWL 2 web ontology language document overview (second edition). W3C Recommendation, 11 December 2012, 2012. URL `https://www.w3.org/TR/owl2-overview/`.

A What limits coverage

On grounded MedHallu, 56% of claims resolve at least one entity but yield no checkable proposition, and 70% resolve an entity at all, so relation extraction rather than vocabulary is the binding constraint there. On Med-HALT the position reverses: 67% resolve no entity, a vocabulary limit, reflecting dental, anatomical and genetic terminology outside a drug-disease-laboratory lexicon. Extending either alone would move coverage little, since an extracted predicate with no axiom over it decides nothing.

The deterministic protocol also changes this accounting: the checks run in code, costing a fixed 0.02 to 0.26 s of CPU, rather than additional round trips to the model.

B Results on the external benchmarks

The external benchmarks were run at different times with different arms, so Table 5 lists every scored run rather than a summary that would compare arms never run on the same auditor. The upper block holds the original arms, run on several auditors before the deterministic protocol existed, and the checks alone. qwen3.5:4b is not among the fourteen auditors of Table 2 because its run on our benchmark lacks the method's arms. The lower block is the only external run of the method with its prompt-matched control; it also includes a second checker, a model judging each claim against its supplied passage.

Three things follow. The checks alone are at chance on all three benchmarks. Tool calling off domain is fragile: llama3.1:8b returned no verdict on 222, 299 and 57 of the 400 claims, and exposing the tools lowered gpt-oss:120b on grounded MedHallu from 0.873 to 0.807 accuracy. And on the one benchmark where the method was run, the prompt-matched control leads it, while the gate still matters: without it the checking arm falls from 0.397 to 0.251.

C The checks alone, per auditor

Table 6 gives the per-auditor comparison summarized in Section 4.2. Pooled over auditors, where the auditor departs from the rule it mostly rejects a flag on a claim that is in fact correct (50 of its 54 overrides were right), and where no check fires, the check results it does see move its verdict the right way 34 times and the wrong way 16.

D Where the gain comes from, per auditor

Figure 3 draws the controls of Table 3 per auditor: the step from the prompt-only control to concept lookup, and the step from lookup to the ontology's checks.

Table 5: External benchmarks, MCC; each benchmark is balanced at 400 claims. Upper block: every run of 200 or more scored claims; dashes mark arms not run. Missing verdicts are scored as errors. Lower block: llama3.1:8b on grounded MedHallu, on the 288 claims every listed arm completed. Regenerated by `external_table.py` and `checks_alone.py --fresh`.

Auditor	Arm	MedHallu	MedHallu grounded	Med-HALT
none (checks alone)	—	−0.054	−0.043	+0.046
qwen2.5:7b	no_tools	+0.123	+0.470	+0.090
	cho_tools	+0.021	+0.431	+0.080
llama3.1:8b	no_tools	+0.155	+0.329	+0.152
	cho_tools	−0.555	−0.685	+0.011
qwen3.5:4b	no_tools	−0.182	−0.043	−0.152
	cho_tools	+0.025	+0.103	+0.111
gpt-oss:120b	no_tools	—	+0.748	—
	no_tools_json	—	+0.649	—
	cho_tools	+0.323	+0.620	+0.552

llama3.1:8b, grounded MedHallu, 288 claims	
no_tools / no_tools_json	+0.347 / +0.486
cho_tools	−0.653 (213 claims with no verdict)
cho_context / without the gate	+0.397 / +0.251
passage entailment check	+0.568

Table 6: Checks alone and what the auditor adds, MCC on each auditor's Table 2 claims. With no model, the checks alone score 0.581 (0.584 and 0.603 on the shorter mistral and qwen3.5:122b claim sets). The rule takes a check's flag where one fires and the auditor's own prompt-only verdict otherwise; the last column is what reading the check results adds over that rule, with a paired bootstrap interval († includes zero).

Auditor	Rule	cho_context	Auditor adds [95% CI]
llama3:8b	0.670	0.772	+0.102 [+0.013, +0.204]
qwen3.8:27b	0.831	0.907	+0.076 [+0.018, +0.147]
qwen2.5:7b	0.751	0.815	+0.064 [+0.014, +0.128]
medgemma:27b	0.812	0.877	+0.065 [+0.014, +0.122]
gemma4:31b	0.851	0.926	+0.075 [+0.024, +0.137]
qwen3.5:122b	0.878	0.949	+0.072 [+0.015, +0.136]
llama3.1:70b	0.786	0.864	+0.078 [+0.016, +0.146]
qwen3.5:35b	0.850	0.916	+0.066 [+0.010, +0.138]
granite4.1:3b	0.768	0.761	−0.006 [−0.053, +0.044]†
granite4.1:30b	0.844	0.932	+0.088 [+0.025, +0.159]
nemotron3:33b	0.833	0.874	+0.041 [−0.009, +0.097]†
gpt-oss:120b	0.870	0.944	+0.074 [+0.022, +0.141]
mistral-medium-3.5	0.866	0.954	+0.088 [+0.005, +0.173]
llama3.1:8b	0.791	0.825	+0.034 [−0.040, +0.112]†

E COST OF A CHECKED CLAIM

Table 7 gives the per-claim latency and energy summarized in Section 4.4, for the same 7B auditor served on device, edge and cloud hardware.

Table 7: Latency and energy for one audited claim, the same 7B auditor on three tiers, under the original prompt and under tool calling (`cho_tools`): medians with idle power subtracted, over 229 to 231 claims on the device and 50 on the edge and cloud tiers. The deterministic pass was not profiled. With tool calls the device is about ten times slower and uses about half the energy of the RTX 4090 and two-thirds that of the H200.

	Latency (s)		Energy (J)	
Tier	no_tools	cho_tools	no_tools	cho_tools
Device, Apple M4	3.28	10.36	44.9	150.7
Edge, RTX 4090	0.44	0.99	129.9	292.6
Cloud, H200	0.39	0.83	100.7	222.4

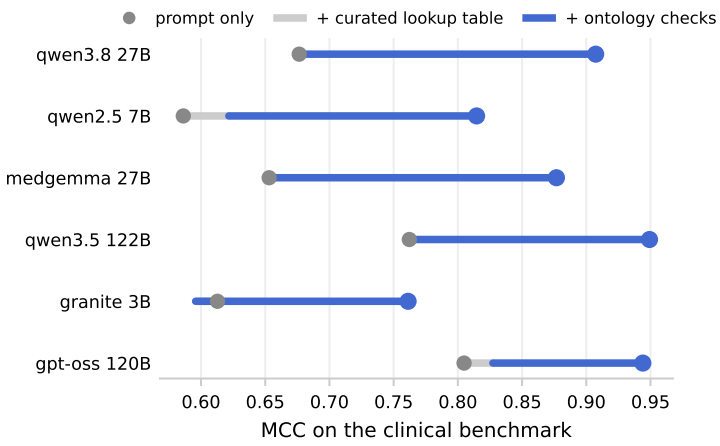


Figure 3: Where the gain comes from, per auditor. The grey segment is what concept lookup over the same vocabulary adds to the prompt-only control; the blue segment is what the ontology's checks add on top. The grey segments are short or absent.

F LABEL REVIEW

The forbidden-relation labels inherit only what the automated mining preserved of the source terminology's context, so a claim generated from a qualified contraindication may describe accepted therapy. Two annotators independently reviewed 110 claims, 100 labelled hallucinated and 10 labelled correct, judging only whether each is clinically correct as written.

No claim labelled hallucinated was judged outright correct by either annotator: 0 of 100, a label error rate with an exact 95% upper bound of 3.6%. Seven of the 100 were judged *context-dependent*: true in some clinical settings and false in others, with the sentence not saying which. These are the cases the mining predicts, and they concentrate on situational contraindications, such as norepinephrine for hypotension, first-line in septic shock but contraindicated in uncorrected hypovolemia. Three of the seven are one claim in different surface forms, so five distinct situations are involved.

Agreement was high, $\kappa = 0.966$ with one disagreement in 110, reflecting that most claims are unambiguous: a serum glucose of 5.0 mg/dL is wrong without reference to context. Two caveats qualify the exercise: the sheet displayed the benchmark's own label, which risks anchoring, though annotators overrode it seven times and no two notes were worded alike; and both annotators are authors rather than independent clinicians. The labels are defensible as written, with about seven percent of those reviewed resting on context the sentence omits.