TaxLogic: Executable Rule Constraints for Rule-Intensive Professional Reasoning

Anonymous authors
Paper under double-blind review

Abstract

Answers in taxation and auditing are governed by explicit written rules, namely statutory rates, arithmetic identities, mandatory evidence, and prescribed procedures. Two bottlenecks arise when language models are deployed in such settings. First, general retrieval augmentation draws on heterogeneous open corpora, so conflicting or outdated provisions are introduced and the uniqueness and authority of the knowledge source required by rigid business processes cannot be guaranteed. Second, conventional knowledge fusion encodes rules implicitly in model parameters, so no hard constraint exists along the inference path and intermediate steps violating the underlying business logic survive into an output that appears plausible while failing the compliance criterion. A two-layer constrained generation paradigm is proposed in which a rule engine is placed upstream of generation and an authoritative textbook corpus serves as the knowledge base. Compliance logic, arithmetic relationships, and required elements are compiled into executable predicates, and each option of a candidate answer must cite either a rule identifier or a textbook page together with a verbatim quotation, with every quotation located against the complete corpus rather than a retrieved fragment. A candidate in which any option fails is voided, and an item that neither source can adjudicate is referred for human review. Closed-form violation bounds are derived for four strategies. Certification and retry-based post-hoc checking are shown to share one lower bound, namely the share of errors the encoding cannot see, and regeneration cannot reach it because a retry carrying a blind-spot violation is accepted and delivered. Two benchmarks with disjoint roles are evaluated. On 231 past CPA examination items, comprising 126 taxation and 105 auditing items with gold labels from the official answer keys, an open 27.9B-parameter model violates the governing rule on 43.7% of items without material, on 18.6% with textbook material only, and on 14.3% with both sources under the citation contract. The reduction over the textbook-only baseline is 4.3 percentage points and is not significant (McNemar $p = 0.14$), a narrowing attributed to the compilation ceiling rather than mechanism failure. Every delivered answer carries a verbatim locatable quotation, 217 of 217 or 100%, while the entry-level location rate is 99.1%, or 874 of 882. On a 360-item rule-derived benchmark that invokes no language model and injects known violations into a shared perturbation stream, TaxLogic retains 6.1% violations against 21.7% for retrieval augmentation and 11.7% for single-pass post-hoc checking, whereas ten retries still retain 8.1%. The architecture is decoupled from the model backbone, requires no parameter fine-tuning, and is offered with a reproducible benchmark framework and an explicitly bounded set of trustworthiness metrics.

1 Introduction

The deployment of large language models in tax compliance, regulatory supervision, and professional practice has accelerated, and the absolute trustworthiness of generated content has become a prerequisite that displaces generalization performance as the primary admis-

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.

sion criterion for domain applications. Such settings follow a principle of zero compromise on rigid rules and zero tolerance for deviating conclusions, since the applicability of a provision and the arithmetic relationships between amounts are defined by publicly issued documents, so an output carrying an implicit knowledge deviation may expose a user to administrative compliance risk or to a tax dispute.

Two dominant technical routes exhibit deficiencies that are structural rather than incidental. The first is retrieval-augmented generation (RAG), in which authoritative policy text is supplied as soft context on the input side while the generative logic remains dominated by implicit parametric memory, so complete coverage of fine-grained rigid rules cannot be enforced and non-authoritative knowledge acquired during pretraining cannot be prevented from participating in generation. Retrieval augmentation of this kind has become a standard baseline for knowledge-intensive tasks (Lewis et al., 2020), yet it has been measured on open-domain question answering that even the strongest models produce citations failing to fully support their statements in approximately 50% of cases (Gao et al., 2023b). The second is post-hoc checking, in which violations are sought only after a complete output is produced and a full regeneration is demanded once a rule triggers. Substantial redundant computation is incurred, the blind spots of the checking module cannot be eliminated, and more fundamentally a retry cannot escape its own lower bound, since a regeneration carrying a blind-spot error passes the checker and is delivered (Section 4). A common shortcoming is that reliance on implicit parametric knowledge is not broken at any point of the pipeline, and a trustworthy generation loop suited to rigid-rule settings has therefore not been formed.

The boundary of prior work. Constraining generation by authoritative text has been pursued along independent lines that have not converged within a single system. Evidence attribution methods generate sentence-level citations but measure anchoring rather than guarantee it, and their citation granularity is a document or passage identifier rather than a verbatim quotation (Gao et al., 2023b; Menick et al., 2022; Rashkin et al., 2023; Zhang et al., 2024; Chuang et al., 2025). Inference-time hard constraints provide provable decoding guarantees, but their constraint languages are hand-designed for a benchmark, are not derived from authoritative documents, and supply no audit trail (Willard & Louf, 2023; Scholak et al., 2021; Park et al., 2024; Banerjee et al., 2025). Step-wise certification issues a certificate for each reasoning step, but the formalisation of a problem from natural language remains the model's own contribution and is not itself certified (Poesia et al., 2024). Authoritative text compilation translates statutes into executable code with a verified core, but the translation step is assumed correct and no theoretical analysis is provided (Merigoux et al., 2021; Jurayj et al., 2026). Formal compliance defines compliance as a machine-checkable relation and proves auditable properties, while excluding free-text output and the correctness of the specification from its scope (Avni et al., 2026; Solatorio, 2025). This work therefore does not claim to introduce rule-constrained generation for the first time. It claims that anchoring each delivered conclusion verbatim in an authoritative rule or textbook, while making every delivered conclusion independently locatable, has been realised as an inference-time hard constraint and supplied with a closed-form derivation of its violation rate, and that this conjunction is not covered by existing work.

Contributions. First, a quantitative model of trustworthy generation is established, in which a mapping among rule-constraint strength, knowledge-source authority, and violation rate is defined and closed-form bounds are derived for four strategies (Propositions 1 to 3). Certification and retry-based post-hoc checking share one lower bound, the share of errors the encoding cannot see, while retries cannot reach it because a retry carrying a blind-spot violation passes the checker and is delivered. Second, a two-layer constrained generation paradigm is proposed, in which heterogeneous professional rules are compiled into executable predicates that run at inference time and a citation contract requires every conclusion to carry a triple of source, location, and verbatim quotation, each quotation being located against the complete corpus rather than merely self-consistent within a retrieved fragment. A quotation that cannot be located voids the answer, and an item that neither source can adjudicate is referred. The architecture is decoupled from the model backbone and requires no retraining when rules are updated. Third, a benchmark aligned with the official professional qualification examination is constructed from past items of the CPA

Taxation and Auditing papers, comprising 231 items with gold labels taken from the official answer key of each paper and therefore independent of the rule library used here. A further 360-item rule-derived benchmark is used for mechanism attribution, so that a reduction in violation rate is decomposed across fact-conditional routing, rule certification, local repair, and referral rather than reported as a single aggregate. The two benchmarks answer different questions and are never combined: the former measures the capability of a method on real examination items, the latter the behaviour of a mechanism under known errors. Traceability is reported at two levels, since conflating them systematically flatters a system. The delivered-item traceability rate counts delivered answers in which every selected option carries a locatable quotation, while the entry-level location rate counts all citation entries including those on unselected options.

2 Related Work

Factuality, constrained decoding, and search. Surveys identify generation departing from source truth as a core reliability failure (Ji et al., 2023), and Lin et al. (2022) show that models reproduce confident falsehoods. Chain-of-verification and FActScore improve or measure the support of generated text (Dhuliawala et al., 2023; Min et al., 2023), but all verify after the fact and none guarantees that a delivered answer satisfies an executable domain constraint. Grammar-constrained decoding restricts each token to a formal language and makes mask computation inexpensive through indexing (Geng et al., 2023; Willard & Louf, 2023; Scholak et al., 2021; Park et al., 2024); that cost is measured here and the difficulty is found not to lie there but in the fact that professional rules are conditional constraints over a case, since which rate applies, whether a stated amount is consistent with it, and whether the mandatory clauses are complete are conditions invisible to a grammar over a set of legal tokens. Self-consistency, self-refinement, tree search, process supervision, and tool-interactive critique improve or repair answers by spending additional inference (Wang et al., 2023; Madaan et al., 2023; Yao et al., 2023a; Lightman et al., 2024; Gou et al., 2024), and the trade-off is made explicit for the verification loop in Section 4.

Neuro-symbolic systems, domain benchmarks, and authoritative compilation. Program-aided language models and Logic-LM delegate computation to an interpreter (Gao et al., 2023a; Pan et al., 2023), DSPy compiles declarative pipelines (Khattab et al., 2024), and multi-step generation is organised by ReAct and chain-of-thought prompting (Yao et al., 2023b; Wei et al., 2022). LegalBench and FinGPT/PIXIU construct legal and financial evaluation sets (Guha et al., 2023; Yang et al., 2023; Xie et al., 2023); in contrast, the evaluation here uses professional qualification examination items carrying official answer keys rather than items generated from the rule library. Catala compiles statutory fragments to a verified core (Merigoux et al., 2021), statutory logic programs compile legislation into a Prolog rule base (Jurayj et al., 2026), and proof-carrying output defines compliance formally with machine-checkable proofs (Avni et al., 2026; Solatorio, 2025); in each case the translation from source text into a formal representation is assumed correct, and none supplies a verbatim citation constraint with a closed-form derivation of the violation rate.

Decoding-time authority: the closest prior art. Arjmandi (2026) apply exact masking at the decoding layer and provide proved guarantees, but the authority invoked is a grammar and an access-control policy rather than a statute or a standard. Hall et al. (2026) compile filing facts into hard Z3 constraints in the accounting domain, but their operands are machine-readable structured facts and prose standards text is never touched. Simpson et al. (2026) author rules from verbatim source text while stating that their guarantees cover only the execution layer, leaving unresolved whether an authored formula faithfully represents the source legislation. Two gaps are therefore quantified rather than asserted: no system enforces verbatim citation of a statute or standard at generation time, and no system grounds against a complete authoritative prose corpus. The design reported here differs on exactly these two points.

3 The TaxLogic Two-Layer Constrained Generation Paradigm

The architecture is summarised in Figure 3 in Appendix G: two authoritative sources are grounded, compiled into predicates, and passed to a single constrained-inference path whose verification gate yields one of three outcomes, namely ship, void, or refer.

Problem formulation. Let x denote a query, $\mathcal{R}$ the complete rule set, F_x the facts extracted from x, and $\mathcal{R}_x = \text{Router}(F_x, \mathcal{R}, K) \subseteq \mathcal{R}$ the active subset. An answer y is a structured object with named fields, comprising a rate declaration, a computation, an evidentiary clause, and a procedural statement, and $H(y \mid \mathcal{R}_x)$ is the hard validity predicate compiled from the active rules. The answer is delivered if and only if $H(y \mid \mathcal{R}_x) = 1$:

$$\text{deliver } y \iff H(y \mid \mathcal{R}_x) = 1, \qquad H(y \mid \mathcal{R}_x) = \prod_{r_i \in \mathcal{R}_x} \Big(1 - \phi_i(F_t) + \phi_i(F_t)\, \rho_i(F_t, y)\Big), \quad (1)$$

where ϕ_i is the trigger predicate and ρ_i the validity predicate. A candidate is proposed by a generator P_θ. When a candidate is rejected, the offending field is re-derived by the engine from the routed rules rather than the model being resampled; the engine does not relax the constraint; and referral is returned when no active row can adjudicate a field. Certification is thus a rejection test over a complete candidate rather than a token-level constrained decoder, so the generator remains unconstrained while the constraint layer decides what may be delivered.

Rules, routing, and incremental constraints. Each source rule, whether a statutory line of tax law, an atomic auditing rule, or an accounting account or indicator rule, is normalised into a nine-tuple comprising identifier, scope, facts, logic, constraints, exceptions, output, and governance. Compilation yields parameter tables, applicability predicates over trigger and exception expressions, and answer schemas, each returning one of Pass, Fail, Data-Incomplete, Legal-Exception, Conflict, or Review; the proportion of each library that can yield such predicates is audited in Section 6.3.

Rules are scored against the extracted facts, with domain, entity, amount, and effective period matched, and the top K above a threshold are retained. Routing plays a dual role: it supplies the context used by RAG-style methods and defines the active set over which the engine evaluates its predicates, so checking cost grows with K rather than with $|\mathcal{R}|$. For a candidate fragment v at step t, let $\mathcal{R}_{\text{aff}}(S_t, v) \subseteq \mathcal{R}_x$ be the set of affected rules, giving the hard validity function

$$H_t(v) = \prod_{r_i \in \mathcal{R}_{\text{aff}}} \Big(1 - \phi_i(F_t) + \phi_i(F_t)\, \rho_i(F_t, v)\Big), \quad (2)$$

so the constrained probability of any candidate violating a triggered rule is zero. A legitimate exception is always an explicit output state and is never silently converted into a pass. Rate attribution, arithmetic consistency, and clause completeness are checked, and reference answers are never consulted.

Certification and local repair. A candidate is certified field by field against the routed rules. A declared rate must belong to the rate rows the router placed in the active set for the tax type of the case; an amount must equal revenue multiplied by the rate within tolerance; and an auditing answer must retain its evidence and full-inventory clauses and cite the atomic rule it executes. Three outcomes are kept distinct, namely ok, which is delivered; violation, in which the engine repairs the offending field; and uncertifiable, in which the active set lacks the rows required for checking and the item is referred. Repair re-derives the field from the routed rules, reading the applicable bracket from the wording of the item alone and reading no label, and neither the checker nor the repair step consults a reference answer. Safety is therefore conditionally dependent on routing in a measurable way, since a case whose governing row was not recalled cannot be certified and can only become a referral.

The citation contract: two sources, one constraint surface. When a question is answered in prose, and equally when the governing material is expository rather than a rate table,

certifying answer fragments alone is insufficient, so a fourth class of object is added. For each assertion c, a triple $(\mathrm{src}(c), \mathrm{loc}(c), \mathrm{quo}(c))$ is required, in which $\mathrm{src}(c) \in \{\mathrm{rule}, \mathrm{textbook}\}$, $\mathrm{loc}(c)$ is a rule identifier or textbook page, and $\mathrm{quo}(c)$ is the quoted text. Delivery requires $H(y \mid \mathcal{R}_x) \cdot H_{\mathrm{cite}}(y) = 1$, where

$$H_{\mathrm{cite}}(y) = \prod_{c \in y} \mathbb{1}\big[\mathrm{loc}(c) \in \mathrm{ids}\big(\mathrm{src}(c)\big)\big] \cdot \mathbb{1}[\mathrm{quo}(c) \sqsubseteq \mathrm{src}(c)]\,, \tag{3}$$

and $\sqsubseteq$ denotes verbatim containment after punctuation and whitespace normalisation. An assertion whose quotation cannot be located voids the entire answer, and an answer in which every assertion declares no applicable source is referred.

Two implementation decisions were forced by observed failures, each silently converting a correct and well-grounded answer into an output that appears fabricated. First, checking is performed against the complete corpus rather than retrieved fragments, since an initial version comparing each quotation with the fragment placed into the prompt voided eight answers on the taxation arm, seven of which stated the correct answer; because quotations legitimately elide and join sentences, a quotation is split on ellipses and sentence-final punctuation and each segment must be located. Second, textbook scans carry a publisher watermark mixed into the text by optical character recognition at mid-sentence positions, and the watermark block was inferred from the data and removed.

4 Theoretical Analysis of the Strategy Space

Let a candidate event be drawn per item position with contamination probability p, let a mutation be detectable with probability $1 - \beta$ and fall in a blind spot with probability β, and let a detectable mutation be flagged by the checker with probability d, measured at $d = 0.9888$. The three arms without a checker retain all contamination they produce, $\mathrm{RV}_a = p_a$ for $a \in \{A, R, B\}$, whereas certification removes a further factor, $\mathrm{RV}_T \le p_T[\beta + (1-\beta)(1-d)]$, so the measured ordering $A > R > B > T$ separates the effect of supplying material, which lowers p_a, from the effect of the constraint layer, which filters the contamination produced.

Proposition 1 (safety of a certified answer). A violation is retained if and only if the mutation lies in a blind spot or is a detectable mutation the checker failed to flag and then accepted, giving $\mathrm{RV}_{\mathrm{TL}} \le p[\beta + (1-\beta)(1-d)]$. The retained rate does not depend on whether repair succeeds, since a repair failing certification becomes a referral rather than a violation.

Proposition 2 (single-pass post-hoc checking). A strategy inspecting a finished answer, performing one independent regeneration at the same contamination rate upon failure, and then delivering the second answer unconditionally retains $\mathrm{RV}_{\mathrm{PH}} = \mathrm{RV}_{\mathrm{TL}} + p^2 d(1-\beta)$, so single-pass post-hoc checking is strictly worse than certification by the term $p^2 d(1-\beta)$.

Proposition 3 (the retry frontier). With at most N independent regenerations, where $s = p\beta + p(1-\beta)(1-d)$ is the probability a draw is accepted and violates and $f = p(1-\beta)d$ the probability a further draw is triggered, the retained rate satisfies $\mathrm{RV}_N = s + fs(1 - f^N)/(1-f) + f^{N+1}$. As $N \to \infty$ the family converges to $\mathrm{RV}_\infty = s/(1-f)$, which exceeds the TaxLogic bound s whenever $s > 0$ and $f > 0$, because a blind-spot retry passes the checker and regeneration reintroduces precisely the class of error the encoding cannot see. More retries therefore narrow the distance to RV_∞ rather than to the TaxLogic bound, and because the gap is small when d is large a narrow margin rather than a decisive advantage is predicted.

Proposition 4 and Corollary 1. Local re-derivation re-reads only routed rows at cost $\mathcal{O}(K)$, whereas regeneration pays for re-retrieval and a complete decode, so the strategies differ not in whether they spend on failure but in which currency they spend. In the degenerate regime $p \to 1$, the certified rate tends to the encoding gap $\beta + (1-\beta)(1-d)$ while the single-pass rate tends to 1; with $\beta = 1/3$ and $d = 0.9875$ the limits are 0.341 and 1.

Remark (routing and abstention are coupled). Certification and re-derivation read only routed rows, so $\Pr(\text{referral}) \geq \Pr(g \notin \mathcal{R}_x)$. In the mechanism study all 51 referrals take this form and fall entirely in the taxation domain, so retrieval quality is part of the safety mechanism rather than a preprocessing detail. Derivations, the treatment of correlated retries, and the closed-form cost expressions are given in Appendix B.

5 Experimental Setup

Three workbooks contain 1,703 rules across 95 sheets, comprising 877 statutory taxation rules with 105 rate parameters, 256 atomic auditing rules, and 570 accounting rules. The second evidence source is an authoritative textbook processed by optical character recognition and cleaning, yielding 705,985 characters over 680 pages for taxation and 663,438 characters over 658 pages for auditing. Both are indexed with weighted BM25 (Robertson & Zaragoza, 2009), a sparse retrieval model also deployed as a baseline in open-domain question answering (Karpukhin et al., 2020); the full weights and the retrieval protocol are given in Appendix A.

Evaluation is conducted on 231 past professional-stage CPA items with gold labels from the official answer key of each paper. The taxation subset contains 126 items, of which 78 are single-choice and 48 multiple-choice; the auditing subset contains 105 items, of which 60 are single-choice and 45 multiple-choice. Forty-six items, or 20%, are negative-polarity items in which option letters are inverted, so letter matching alone is insufficient, and scoring is exact match against the official key. This benchmark replaces a 360-item set built from rule rows, whose labels were programmatic and which could not test applicability, exceptions, or negative polarity since the workbooks carry no executable exclusion or threshold conditions. The 360-item set is retained for the controlled study, and the present revision covers taxation and external auditing.

Material conditions. Arm A supplies no material, arm R only routed rule rows, arm B only routed textbook passages, and arm T both under the citation contract. Arm R is the attribution arm, so the gain of T over A is decomposed into a rule layer, a textbook layer, and their interaction rather than resting on citation counts alone. No prompt contains an answer, verified mechanically over all 693 records by scanning for answer markers, confirming the stem is present, and confirming option letters are never annotated; rules are indexed by content rather than identifier and title. In the controlled comparison one candidate event is drawn per item position, carrying a mutation with probability p and otherwise being the reference solution, with nine mutation classes covering two detectable classes per domain plus one blind spot outside the encoding. All retrieval-dependent methods replay the same stream, and Vanilla draws independently at $p = 0.42$ while the others use $p = 0.20$. The study runs on the 360-item rule-derived benchmark, whose perturbation stream is defined over rate and amount fields; letter-answer items contain no such fields, so results are marked rule-derived and never mixed with the CPA figures.

5.1 Two Benchmarks with Disjoint Roles

Two benchmarks are used and their figures are never combined, because they answer different questions. The first comprises 231 CPA items with official answer keys, which measure what a deployed system encounters but cannot explain why a method works, since errors on real items are produced by the model itself and are therefore neither controllable nor reproducible; an answer to a multiple-choice item is also a letter, and a letter carries no rule violation that could be programmatically injected or detected. The second is the 360-item rule-derived benchmark used for mechanism attribution, whose field-level answers permit known violations to be injected into a stream shared by all strategies, so that any difference among strategies can originate only in the strategies themselves. Mechanism attribution denotes the decomposition of a reduction in violation rate across fact-conditional routing, certification, local repair, and referral rather than a single aggregate improvement. No model is invoked in the second study, deliberately, because a model call would introduce sampling variance and contaminate the decomposition; its figures therefore represent no model capability and are never merged with the CPA results. The per-answer engine overhead reported

Table 1: Real-model arms over 231 past CPA items under four material conditions. Violations are measured against the official answer key rather than injected. The violation rate is the share of wrong answers over valid items, with referrals and voids counted in the denominator and not the numerator, so an arm cannot lower the metric by refusing to answer. Arm R isolates the rule layer and serves as the attribution reference for the gain of T over A.

Arm	Material	Taxation n=126	Auditing n=105	Combined n=231	Referral	Void
A	none	42.9	44.8	43.7	0	0
R	rules only	38.1	36.2	37.2	0	0
B	textbook	21.4	15.2	18.6	0	0
T	rules and textbook (TaxLogic)	19.8	7.6	14.3	7	7

is the wall-clock time of the router and checker in Python and is not inference latency. The design rationale and the internal-validity limitations of the injection stream are given in Appendix A.

5.2 Real-Model Arms and Metrics

An open-source 27.9B-parameter instruction model is run on all 231 CPA items under the four material conditions, comprising 2,541 calls, and each strategy is applied to the same set of real generations before scoring against the official answer key. The decision of each strategy is made by the checker, and the scorer evaluates only the answer a strategy finally delivers, so no strategy observes the label against which it is scored. The delivered answers, per-option verdicts, prompts, and scoring code are released with the artefact. Vanilla denotes unconstrained generation, Keyword-RAG appends retrieved rules without a checker, Post-hoc checks and regenerates, Guided decoding constrains the rate field to a legal enumeration, and TaxLogic performs routing, certification against routed rows and the citation contract, local re-derivation, and referral. On the CPA arms the violation rate is the share of delivered answers disagreeing with the official answer key over valid items, with referrals and voids counted in the denominator and not the numerator, so an arm cannot improve the metric by refusing to answer; referral and void rates are reported separately and never offset against one another. Traceability is reported at two levels, since conflating them flatters a system: the delivered-item traceability rate counts delivered answers in which every selected option carries a locatable quotation, and the entry-level location rate counts all citation entries including those on unselected options. Two of the five strategies do not run on CPA items, namely Post-hoc, which is defined over the controlled study, and Guided decoding, which constrains a rate field that a letter-answer item does not possess; the CPA comparison is accordingly among four material conditions, and the five-strategy comparison is conducted over the controlled study. Full details of the strategy definitions and the statistical protocol are given in Appendix C.

6 Results

6.1 Violations Measured on Real CPA Items

Table 1 reports the primary result over all 231 CPA items scored against the official answer key. Without material the model violates the governing rule on 43.7% of items, routed textbook material reduces this to 18.6%, and 14.3% is reached when both sources are supplied under the citation contract, the lowest of the four conditions, delivering 217 of the 231 items.

The gain is not carried by rules alone. Arm R isolates the rule layer: rules alone without textbook material reduce the violation rate from 43.7% to 37.2%, comprising 42.9% to 38.1% on taxation and 44.8% to 36.2% on auditing, a gain of 6.5 percentage points. Textbook material alone reaches 18.6%, and adding rules reaches 14.3%. Both sources carry signal,

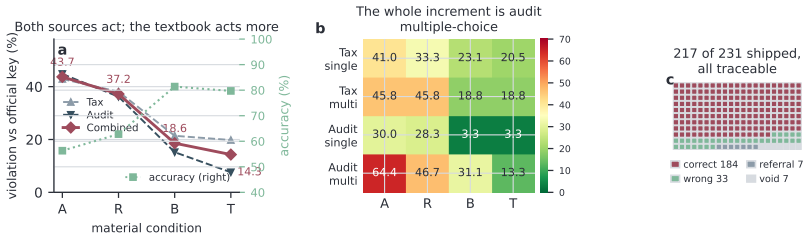

Figure 1: Primary real-model result. (a) Violation rate on the left axis and accuracy on the right axis across the four material conditions. The violation rate falls monotonically from 43.7% to 14.3%, while accuracy is not monotone, the textbook-only arm being highest at 81.4% and TaxLogic reaching 79.7%. (b) A heat map over item type and material condition; auditing multiple-choice items fall from 64.4% to 13.3% and account for the whole increment. (c) A per-item grid for 217 of 231 items, including seven referrals and seven voids.

but on examination items the textbook performs most of the work and the compiled rules add only a modest increment.

The margin narrows, and the compilation ceiling explains it. In the controlled study TaxLogic is placed 15.6 percentage points below keyword RAG; on real items the margin over the textbook-only arm is 4.3 percentage points and is not significant, with a paired McNemar test giving $p = 0.14$. This is what the compilation audit predicts, since the difficulty of an examination item lies in applicability and the compilation layer can certify only what it can compile. The next increment therefore lies in compiling applicability rather than in retrieval.

The auditing violation rate falls to 7.6% against 15.2% and the taxation rate to 19.8% against 21.4%, so TaxLogic attains the lowest rate in both subjects. The auditing gain is nevertheless confined to multiple-choice items, falling from 31.1% to 13.3%, a reduction of 17.8 percentage points, whereas single-choice items already stand at 3.3% on both arms, since compiled clause predicates catch clauses a multi-option answer omits or inverts and add almost nothing where retrieval is already sufficient. Taxation multiple-choice items stand at 18.8% on both arms, and the whole taxation increment lies in single-choice items, falling from 23.1% to 20.5%. Seven items are referred because neither source can adjudicate them and seven are voided because a quoted passage cannot be located.

Every answer delivered by TaxLogic carries a quotation located verbatim in the source it declares, giving 217 of 217 or 100%, while the location rate for individual citation entries is 99.1%, or 874 of 882, the eight failing entries falling on unselected options or on items whose contract was voided. Traceability is not correctness, since a wrong answer remains traceable to the provision on which the model relied.

6.2 Mechanism Comparison and the Retry Frontier

Every figure in this subsection is measured on the 360-item rule-derived benchmark under a shared candidate event stream, is not obtained on CPA items, and is never subtracted from the figures of Table 1. The perturbation stream applies mutations to rate and amount fields, and letter-answer items contain no such fields, so every retrieval-dependent strategy replays exactly the same contaminated candidates and context, and differences isolate the mechanism alone.

TaxLogic delivers 6.1% violations against 21.7% for keyword RAG and 41.4% for unconstrained generation. Retries improve the figure only from 8.6% at $N = 1$ to 8.1% at $N \geq 3$ and never reach 6.1%, in agreement with Proposition 3. The retry frontier is shown in Appendix G and the complete table in Appendix F. Routing determines which cases can be adjudicated and the source of referrals; certification removes detectable mutations from the delivered set; local repair corrects a violating field without a re-decode, with cost separation given by Proposition 4; and referral handles cases neither source can adjudicate, with the bound given by Remark 2. No claim about model capability is made from this subsection.

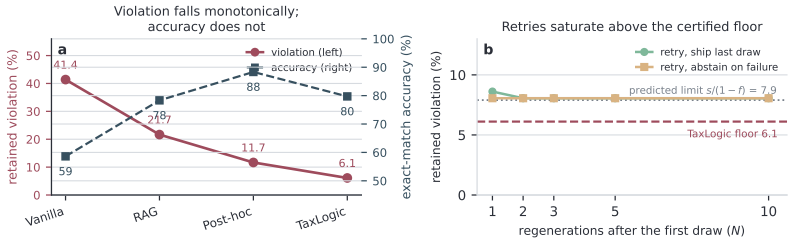

Figure 2: Primary result of the controlled mechanism attribution over the 360-item rule-derived benchmark, in which no language model is invoked. (a) Violation rate on the left axis as a solid line and exact-match accuracy on the right axis as a dashed line. The violation rate falls monotonically from 41.4% to 21.7%, to 11.7%, and to 6.1%, while accuracy is not monotone, being highest for Post-hoc at 88.3% and 79.7% for TaxLogic. (b) The retry saturation curve; between one and ten retries the violation rate remains stable at 7.9% and never reaches the 6.1% floor, in agreement with Proposition 3.

6.3 Retrieval Quality and the Safety Ceiling

Two boundary conditions are measured rather than assumed. First, Guided decoding cannot run on CPA items at all, since it constrains a rate field to a legal enumeration whereas masking option letters is the identity operation; where it does apply it retains 18.6% violations against 6.1% for TaxLogic. Second, every safety claim is bounded by the proportion of the rule library that can be made executable. All 1,703 rows parse and 100% agree with the tax-type field, but only 62 of 877 statutory taxation rules carry a statutory rate, or 7.1%, fifteen carry a formula, and none carries an exclusion condition, so the ceiling is a property of the encoding rather than of the search. Retrieval coverage under content-based indexing is reported in Appendix E.

7 Discussion

Two findings qualify the results. The encoding ceiling is measured rather than assumed, since real examination items remove the parameter-table restriction of the controlled study and the margin over the textbook-only arm falls from 15.6 to 4.3 percentage points and is not significant, which locates the next increment in the fields the rule library does not carry. The checker is itself inside the threat model, since four defects in the checker used here each converted a legitimate citation into an output that appeared fabricated and raised the apparent violation rate by approximately 5 percentage points, with failures that are silent. The impossibility results, the internal validity of the rule-derived benchmark, and the full statement of limitations are given in Appendix D.

8 Conclusion

TaxLogic certifies answers against constraints compiled from the rules recalled by the router and against verbatim quotations from the textbook explaining them. A field alone is repaired on failure, the whole answer is voided when a quotation cannot be located, and referral is issued when neither source can adjudicate. Certification keeps detected violations out of the delivered set without requiring regeneration, and retries cannot substitute for it, since both share the same encoding gap and a retry stops at a higher plateau. On the 231-item CPA benchmark the method retains 14.3% violations against 18.6% for textbook-only retrieval and 43.7% with no material, while supplying a locatable citation for every answer delivered. What is offered is therefore a reproducible benchmark, a derivable bound, and an explicitly bounded set of metrics rather than an assertion of greater trustworthiness.

AI use statement

(This section is required and does not count toward the page limit.)

Generative AI tools were used for tasks in the disclosure-required category, namely generating the synthetic rule-grounded benchmark and the contamination simulation, implementing the experimental pipeline and the TaxLogic prototype, cleaning and restructuring the rule dataset and textbook corpus, extracting the past CPA items constituting the benchmark, and assisting in the interpretation of results. They were not used to fabricate data or to determine the reported measurements or the conclusions. They were additionally used for tasks in the recommend-disclosure category, namely producing scientific figures, drafting parts of the manuscript, surveying and identifying relevant literature, and suggesting paper structure. All AI-assisted work has been reviewed. Every numerical claim is produced by the released deterministic pipeline and re-verified against machine-readable output. Real-model generations are produced by the released harness and scored by the released rule-grounded scorer, and item scoring is performed by a language model acting as judge, which is explicitly not human expert review. Every reference has been checked entry by entry against its primary source. The mathematical propositions were derived with AI assistance and checked line by line. Responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI, is taken by the authors.

Reproducibility statement

(This section is recommended and does not count toward the page limit.)

The complete source code, the three rule workbooks, the benchmark items with official gold labels, the cleaned textbook corpora, the per-item result records, and the figure-generation scripts are released as an anonymous supplementary archive. The pipeline that produces every reported quantity is deterministic, and each figure and table is regenerated from the machine-readable records held under results/. The four real-model arms are produced by the harness in code/taxlogic_v4/run_v4.py under fixed generation settings, namely temperature 0 and a fixed seed, and are scored by exact match against the official answer keys with referrals and voids retained in the denominator. The 360-item controlled study in code/cindy2/ invokes no language model and is fully deterministic, so its figures are reproduced exactly without any model access. Assumptions and the complete derivations for Propositions 1 to 4 and Corollary 1 are given in Appendix B, the strategy definitions and the statistical protocol in Appendix C, the complete per-arm records of the controlled study in Appendix F, and the measurement procedure for the encoding ceiling in Appendix E.

Ethics statement

(This section is recommended and does not count toward the page limit.)

The benchmark consists of past professional examination items that are publicly available, together with their official answer keys, and no human subjects were involved, so no institutional review was required. The rule workbooks and the textbook corpora are used as authoritative reference material and are redistributed only within the supplementary archive. The system is designed to support rather than to replace professional judgement: it issues a referral when no source can adjudicate, it reports the traceability of a delivered answer separately from its accuracy, and it is not intended for use as a substitute for a qualified practitioner. Because a delivered answer may still be wrong, the contract is presented as making a wrong answer reviewable rather than merely unusable, and the measured encoding ceiling is stated explicitly so that the guarantee is not overstated. Taxation and auditing carry legal and financial consequences, and the negative results reported here, in particular the 4.3 percentage point margin that does not reach significance on real items, are stated in full rather than omitted.

References

Mohsen Arjmandi. GRID: Grammar-railed decoding for enterprise SQL generation. arXiv preprint arXiv:2607.11951, 2026.

Hillel Avni, Shlomi Dolev, Avraam Yagudaev, and Moti Yung. Super-intelligence survival guide: Verification via proof-carrying output. IACR Cryptology ePrint Archive, Paper 2026/994, 2026. URL https://eprint.iacr.org/2026/994.

Debangshu Banerjee, Tarun Suresh, Shubham Ugare, Sasa Misailovic, and Gagandeep Singh. CRANE: Reasoning with constrained LLM generation. In Proceedings of the 42nd International Conference on Machine Learning (ICML), volume 267 of PMLR, pp. 2836–2857, 2025.

Yung-Sung Chuang, Benjamin Cohen-Wang, Shannon Zejiang Shen, et al. SelfCite: Self-supervised alignment for context attribution in large language models. In Proceedings of the 42nd International Conference on Machine Learning (ICML), 2025.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, et al. Chain-of-verification reduces hallucination in large language models. arXiv preprint arXiv:2309.11495, 2023.

Luyu Gao, Aman Madaan, Shuyan Zhou, et al. PAL: Program-aided language models. In Proceedings of the 40th International Conference on Machine Learning (ICML), volume 202 of PMLR, pp. 10764–10799, 2023a.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 6465–6488, 2023b. doi: 10.18653/v1/2023.emnlp-main.398.

Saibo Geng, Martin Josifoski, Maxime Peyrard, and Robert West. Grammar-constrained decoding for structured NLP tasks without finetuning. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 10932–10952, 2023.

Zhibin Gou, Zhihong Shao, Yeyun Gong, et al. CRITIC: Large language models can self-correct with tool-interactive critiquing. In International Conference on Learning Representations (ICLR), 2024.

Neel Guha, Julian Nyarko, Daniel E. Ho, et al. LegalBench: A collaboratively built benchmark for measuring legal reasoning in large language models. In Advances in Neural Information Processing Systems 36 (NeurIPS), 2023.

Bethel Hall, Sachi Shome, and William Eiers. VeriFin: A neurosymbolic framework for verifying LLM-generated financial claims. arXiv preprint arXiv:2608.10213, 2026.

Ziwei Ji, Nayeon Lee, Rita Frieske, et al. Survey of hallucination in natural language generation. ACM Computing Surveys, 55(12):1–38, 2023. doi: 10.1145/3571730.

William Jurayj, Nils Holzenberger, and Benjamin Van Durme. Language models and logic programs for trustworthy tax reasoning. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 40, pp. 38688–38698, 2026. doi: 10.1609/aaai.v40i45.41212.

Adam Tauman Kalai and Santosh S. Vempala. Calibrated language models must hallucinate. In Proceedings of the 56th Annual ACM Symposium on Theory of Computing (STOC), pp. 160–171, 2024. doi: 10.1145/3618260.3649777.

Michał P. Karpowicz. On the fundamental impossibility of hallucination control in large language models. arXiv preprint arXiv:2506.06382, 2025.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, et al. Dense passage retrieval for open-domain question answering. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 6769–6781, 2020. doi: 10.18653/v1/2020.emnlp-main.550.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, et al. DSPy: Compiling declarative language model calls into state-of-the-art pipelines. In International Conference on Learning Representations (ICLR), 2024.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In Advances in Neural Information Processing Systems 33 (NeurIPS), pp. 9459–9474, 2020.

Hunter Lightman, Vineet Kosaraju, Yura Burda, et al. Let's verify step by step. In International Conference on Learning Representations (ICLR), 2024.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL), pp. 3214–3252, 2022.

Aman Madaan, Niket Tandon, Prakhar Gupta, et al. Self-refine: Iterative refinement with self-feedback. In Advances in Neural Information Processing Systems 36 (NeurIPS), 2023.

Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. Psychometrika, 12(2):153–157, 1947. doi: 10.1007/BF02295996.

Jacob Menick, Maja Trebacz, Vladimir Mikulik, et al. Teaching language models to support answers with verified quotes. arXiv preprint arXiv:2203.11147, 2022.

Denis Merigoux, Nicolas Chataing, and Jonathan Protzenko. Catala: A programming language for the law. Proceedings of the ACM on Programming Languages, 5(ICFP):1–29, 2021. doi: 10.1145/3473582.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, et al. 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 (EMNLP), pp. 12076–12100, 2023.

Liangming Pan, Alon Albalak, Xinyi Wang, and William Wang. Logic-LM: Empowering large language models with symbolic solvers for faithful logical reasoning. In Findings of the Association for Computational Linguistics: EMNLP 2023, pp. 3806–3824, 2023.

Kanghee Park, Jiayu Wang, Taylor Berg-Kirkpatrick, Nadia Polikarpova, and Loris D'Antoni. Grammar-aligned decoding. In Advances in Neural Information Processing Systems 37 (NeurIPS), 2024.

Gabriel Poesia, Kanishk Gandhi, Eric Zelikman, and Noah D. Goodman. Certified deductive reasoning with language models. Transactions on Machine Learning Research (TMLR), 2024. arXiv:2306.04031.

Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, et al. Measuring attribution in natural language generation models. Computational Linguistics, 49(4):777–840, 2023. doi: 10.1162/coli_a_00486.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. Foundations and Trends in Information Retrieval, 3(4):333–389, 2009. doi: 10.1561/1500000019.

Torsten Scholak, Nathan Schucher, and Dzmitry Bahdanau. PICARD: Parsing incrementally for constrained auto-regressive decoding from language models. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 9895–9901, 2021.

Paul Simpson, John Kozak, and Lisa Doake. Confidently wrong: Exception chain collapse in frontier LLM rule evaluation. arXiv preprint arXiv:2607.23386, 2026.

Aivin V. Solatorio. Proof-carrying numbers (PCN): A protocol for trustworthy numeric answers from LLMs via claim verification. arXiv preprint arXiv:2509.06902, 2025.

Xuezhi Wang, Jason Wei, Dale Schuurmans, et al. Self-consistency improves chain of thought reasoning in language models. In International Conference on Learning Representations (ICLR), 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, et al. Chain-of-thought prompting elicits reasoning in large language models. In Advances in Neural Information Processing Systems 35 (NeurIPS), pp. 24824–24837, 2022.

Frank Wilcoxon. Individual comparisons by ranking methods. Biometrics Bulletin, 1(6): 80–83, 1945. doi: 10.2307/3001968.

Brandon T. Willard and Rémi Louf. Efficient guided generation for large language models. arXiv preprint arXiv:2307.09702, 2023.

Qianqian Xie, Weiguang Han, Xiao Zhang, et al. PIXIU: A comprehensive benchmark, instruction dataset and large language model for finance. In Advances in Neural Information Processing Systems 36 (NeurIPS), 2023.

Ziwei Xu, Sanjay Jain, and Mohan Kankanhalli. Hallucination is inevitable: An innate limitation of large language models. arXiv preprint arXiv:2401.11817, 2024.

Hongyang Yang, Xiao-Yang Liu, and Christina Dan Wang. FinGPT: Open-source financial large language models. arXiv preprint arXiv:2306.06031, 2023.

Shunyu Yao, Dian Yu, Jeffrey Zhao, et al. Tree of thoughts: Deliberate problem solving with large language models. In Advances in Neural Information Processing Systems 36 (NeurIPS), 2023a.

Shunyu Yao, Jeffrey Zhao, Dian Yu, et al. ReAct: Synergizing reasoning and acting in language models. In International Conference on Learning Representations (ICLR), 2023b.

Jiajie Zhang, Yushi Bai, Xin Lv, et al. Longcite: Enabling LLMs to generate fine-grained citations in long-context QA. arXiv preprint arXiv:2409.02897, 2024.

A Two Benchmarks and Mechanism Attribution

The 231 items carry official answer keys independent of the rule library used here, verified mechanically in that item stems appear zero times in the rule library and answer markers never appear. They therefore measure what a deployed system encounters, namely real applicability judgements, real exceptions, and negative-polarity items, but they cannot explain why a method works. Errors on real items are produced by the model itself and are neither controllable nor reproducible, since it cannot be specified which item position should produce which error, and it cannot be guaranteed that two strategies face the same error. Any observed improvement is therefore confounded with item difficulty. A second obstacle is representational, namely that the answer to a multiple-choice item is a letter carrying no rule violation that could be programmatically injected or detected.

To turn the question of where an improvement comes from into a measurable one, 360 structured items are synthesised from the rule workbooks, with 120 items per domain. Each contains a stem, a reference solution, a rule provenance, and a field-level answer comprising rate, amount, evidence requirement, and scan scope rather than option letters. It is this field structure that makes injection possible, since a violation can be planted in a named field and detected in the same field. It is stated explicitly that the labels are generated programmatically by the rule library and have not been annotated by experts, so the benchmark does not measure professional capability and is used only for controlled comparison among mechanisms.

Mechanism attribution decomposes a reduction in violation rate across the mechanisms producing it, namely fact-conditional routing, certification, local repair, and referral, rather than reporting a single aggregate improvement. The method is to inject known errors rather than wait for the model to err, so each item position receives one of nine mutations with probability p, comprising two detectable classes per domain and one blind spot outside the encoding, and the four strategies replay exactly the same perturbation stream. Because every strategy meets the same error at the same item position, any difference among them can originate only in the strategies themselves.

The absence of any model call is a deliberate design requirement rather than an omission. Any model call introduces a second source of variance, namely the sampling noise of the generator, and would contaminate the decomposition the study is intended to perform. Fixing the generator to a known injection process is what makes the comparison clean. The cost is that these figures represent no model capability, which is exactly why they are never merged with the CPA results. In particular, the per-answer engine overhead reported is the wall-clock time of a Python router and checker and is not inference latency. Its purpose is to test the closed-form analysis of Section 4, and the claim that the method works in practice is carried entirely by the CPA benchmark.

The labels of the controlled benchmark are generated from the programmatic designated values of the rule library, and the fields carrying those values also exist in the retrievable rule rows while the executable checker reads the same column, so the statement that a delivered answer agrees with the rules carries an element of tautology, since a path invoking no model and re-deriving directly from the rule rows would pass the check. This was confirmed empirically during development, in that removing the answer key from the retrievable corpus caused the reference implementation to fail outright when reading that field. The implementation has since been corrected, but annotation of the gold standard by independent experts has not been completed, so it is not claimed that the benchmark has been validated by experts. The study is nevertheless reported, because its purpose is to test the analytical derivation and that purpose does not require the gold standard to be independent of the rule library. The derivation characterises what each strategy would deliver under a given rule representation and injection process, and its correctness is supported by agreement between the derivation and the simulation rather than the provenance of the gold standard. Accordingly the figures may be read only as relative behavioural differences among strategies under controlled conditions, must not be read as model capability, and must not be merged with any figure on the CPA benchmark. Every claim of external validity is carried

by the 231-item CPA benchmark, whose answer keys come from the official answer file and are independent of the rule library used here.

Retrieval settings are as follows. Weighted BM25 is applied over CJK bigrams at weight 1.0, single characters at weight 0.35, ASCII words at weight 1.4, and numeric rate tokens at weight 1.2; rules are indexed by content rather than by identifier and title.

B Derivations for the Strategy Space

The full derivation is reproduced here; the main text states the propositions only.

Let a candidate event C be drawn per item position with $\Pr(C{=}1) = p$, and let a mutation be detectable with probability $1-\beta$ and fall in a blind spot with probability β. A detectable mutation is flagged by the checker with probability d, measured at $d = 0.9888$, while a blind-spot mutation passes by construction. A material condition changes the contamination rate itself, so the contamination rate of arm a is written p_a. The three arms without a checker deliver all contamination they produce, and certification removes a further factor from the remainder:

$$\mathrm{RV}_a = p_a \;\; (a \in \{A, R, B\}), \qquad \mathrm{RV}_T \le p_T\big[\beta + (1-\beta)(1-d)\big]. \tag{4}$$

The measured ordering $A > R > B > T$ separates two effects: material lowers p_a, and the constraint layer removes a further factor from the contamination produced.

Proposition 1 (safety of a certified answer). A violation is retained if and only if the mutation is a blind spot or is a detectable mutation the checker failed to flag and then accepted:

$$\mathrm{RV}_{\mathrm{TL}} \le p\big[\beta + (1-\beta)(1-d)\big], \qquad \text{Abstain} \ge p(1-\beta)\, d\, u. \tag{5}$$

The retained rate does not depend on whether repair succeeds, since a repair failing certification becomes a referral rather than a violation.

Proposition 2 (single-pass post-hoc checking). A strategy inspecting a finished answer, performing one independent regeneration at the same contamination rate p upon failure, and then delivering the second answer unconditionally retains

$$\mathrm{RV}_{\mathrm{PH}} = p\beta + p(1-\beta)\big[(1-d) + d\,p\big] = \mathrm{RV}_{\mathrm{TL}} + p^2 d(1-\beta). \tag{6}$$

A loop regenerating until the checker accepts is a stronger member of the same family and is the comparison a practitioner would adopt, so the whole family is characterised below.

Proposition 3 (the retry frontier). Consider checking with at most N independent regenerations after the first draw, the final draw being delivered if none is accepted. Let

$$s = p\beta + p(1-\beta)(1-d), \qquad f = p(1-\beta)d, \tag{7}$$

so s is the probability a single draw is accepted and violates and f the probability it is rejected and a further draw triggered. Conditioning on an initial rejection, with the final draw delivered with certainty, gives $V(0) = 1$, $V(N) = s + fV(N-1)$ and $\mathrm{RV}_N = s + fV(N)$, solved by

$$\mathrm{RV}_N = s + \frac{fs(1-f^N)}{1-f} + f^{N+1}, \qquad \mathbb{E}[\text{decodes}] = 1 + \frac{f(1-f^N)}{1-f}. \tag{8}$$

As $N \to \infty$ the family converges to $\mathrm{RV}_\infty = s/(1-f)$, which exceeds the TaxLogic bound s whenever $s > 0$ and $f > 0$, because a blind-spot retry passes the checker and regeneration thus reintroduces precisely the class of error the encoding cannot see. More retries therefore narrow the distance to RV_∞ rather than to the TaxLogic bound. The gap is small when d is large, so a narrow margin rather than a decisive advantage is predicted, and this is what is measured.

Remark 1 (correlated retries). Proposition 3 assumes independence. A realistic loop resamples the same model on the same prompt, so retries are positively correlated, the recursion becomes $V(N) = s + \rho V(N-1)$ with $\rho \ge f$, and independence is the optimistic case. The measured direction agrees: retries fail to reduce violations at all.

Proposition 4 (cost separation of repair). Let $\tau_r(K)$ be routing cost at pool cut K, τ_c the cost of one check over the active set, $\tau_{\text{rep}}(K)$ the cost of re-deriving one field, and τ_{dec} the cost of one full regeneration decode, with $\tau_{\text{dec}} \gg \tau_c$. The expected engine cost per case is

$$\mathbb{E}[T_{\text{TL}}] = \tau_r(K) + \tau_c \mathbb{E}[n_{\text{chk}}] + p(1-\beta)d \cdot \tau_{\text{rep}}(K), \tag{9}$$

$$\mathbb{E}[T_{\text{PH}}] = \tau_r(K) + \tau_c + p(1-\beta)d \cdot \big(\tau_r(K) + \tau_{\text{dec}}\big). \tag{10}$$

Local re-derivation re-reads only routed rows at cost $\mathcal{O}(K)$, whereas regeneration pays for re-retrieval and a complete decode, so the strategies differ not in whether they spend on failure but in which currency they spend.

Corollary 1 (degenerate regime). As $p \to 1$, $\text{RV}_{\text{TL}} \to \beta + (1-\beta)(1-d)$, the encoding gap, whereas the single-pass $\text{RV}_{\text{PH}} \to 1$, since a regeneration is almost certain to remain contaminated. With $\beta = 1/3$ and $d = 0.9875$ the limits are 0.341 and 1.

Remark 2 (routing and abstention are coupled). Certification and re-derivation read only routed rows, so a case cannot be adjudicated when $g \notin \mathcal{R}_x$:

$$\Pr(\text{referral}) \geq \Pr(g \notin \mathcal{R}_x). \tag{11}$$

In the mechanism study all 51 referrals take this form and fall entirely in the taxation domain, so retrieval quality is part of the safety mechanism rather than a preprocessing detail.

C Strategy Definitions and Statistical Protocol

Which strategies run where. Post-hoc is defined over the controlled study and reported there, while the CPA arms vary the material condition instead, and arm B is not one of the five strategies. Guided decoding cannot run on CPA items at all, because it constrains a rate field to a legal enumeration whereas a letter-answer item has no rate field, and masking option letters is the identity operation since every letter is legal. The CPA comparison is accordingly among four material conditions, and the five-strategy comparison is conducted over the controlled study. The two tables are labelled separately and are never merged.

Traceability at two levels. Traceability is reported at two levels, since conflating them flatters a system: the delivered-item traceability rate is the share of delivered answers in which every selected option carries a locatable quotation, and the entry-level location rate counts all citation entries including those on unselected options.

Statistical protocol. Comparisons among material conditions use the paired exact McNemar test (McNemar, 1947) on the 231 common items, together with 95% bootstrap confidence intervals over 2,000 resamples for the controlled study and the Wilcoxon signed-rank test (Wilcoxon, 1945) where a continuous per-item quantity is compared.

D Full Statement of Limitations

The encoding ceiling is measured rather than assumed. Real examination items remove the parameter-table restriction under which the controlled study operates, and its margin of 15.6 percentage points becomes 4.3 percentage points and is not significant, as shown in Table 1. This negative result locates the next increment: the constraint layer cannot certify what it cannot compile, and the audit shows the executable payload is thin precisely where examination items probe.

Checker defects and their effect. The checker is itself inside the threat model. Four defects in the checker used here each converted a legitimate citation into an output that appeared fabricated and raised the apparent violation rate by approximately 5 percentage points. Their failures are silent, in that an erroneous field name disables the protection without raising an error while the system still reports a confident and unsupported answer. A

system of this kind should be equipped with a regression suite for its checkers, and the suite used here is released with the artefact. The delivered-item traceability rate of 100% is therefore reported alongside the accuracy of 79.7% rather than combined with it, since the contract promises not that an answer is correct but that a wrong answer names the provision on which the model relied, which makes a wrong answer reviewable rather than merely unusable.

Impossibility results that bound any such system. A calibrated language model must hallucinate on arbitrary facts (Kalai & Vempala, 2024), and no computable language model can learn all computable functions (Xu et al., 2024). More precisely, what a model can be certified to provide is support rather than factual truth (Karpowicz, 2025). These results do not obstruct rule-application tasks, which is the essential point, since a guarantee of complete anchoring cannot originate in a model and can originate only in a constraint on what may be delivered. It follows that zero retained violations means zero under the encoded checks, and that a formal proof guarantees that the proof is valid rather than that the formalisation is faithful.

Internal validity of the rule-derived benchmark. The labels of the controlled benchmark are generated from the programmatic designated values of the rule library, and the fields carrying those values also exist in the retrievable rule rows while the executable checker reads the same column, so the statement that a delivered answer agrees with the rules carries an element of tautology, since a path invoking no model and re-deriving directly from the rule rows would pass the check. This was confirmed empirically during development, in that removing the answer key from the retrievable corpus caused the reference implementation to fail outright when reading that field. The implementation has since been corrected, but annotation of the gold standard by independent experts has not been completed, so it is not claimed that the benchmark has been validated by experts. The study is nevertheless reported, because its purpose is to test the analytical derivation and that purpose does not require the gold standard to be independent of the rule library. Accordingly the figures may be read only as relative behavioural differences among strategies under controlled conditions, must not be read as model capability, and must not be merged with any figure on the CPA benchmark. Every claim of external validity is carried by the 231-item CPA benchmark.

Scope and limitations. The benchmark covers 231 items across two domains, comprising 126 taxation and 105 external auditing items, and is drawn from one examination system in one language. Seventy-five accounting items with official answer keys are held but deliberately excluded. Scoring is exact match against the official answer key, which is a proxy for professional correctness rather than a measure of it. The traceability guarantee is a property of the citation contract rather than of the answer, since quotations are located against the post-OCR corpus and OCR quality is therefore an input. The mechanism study is a controlled contamination model over rule-derived items, and its figures are never merged with the CPA results. Guided decoding does not apply to letter-answer items. The checker is incomplete within its own scope, and the routing diagnostic does not yet define a governing row for items that span several provisions. Zero retained violations means zero under the encoded checks, and the system is not intended to replace professional judgement.

E Retrieval Coverage and Ablations

Indexing rules by content rather than by identifier and title raises key-point coverage from 62.4% to 81.9% on taxation and from 35.0% to 41.2% on auditing. The auditing figure remains low because auditing items ask about concepts whereas the auditing rule library is procedural, a gap that no indexing scheme can close.

Measurement of the safety ceiling is as follows. Every safety claim is bounded by the proportion of the rule library that can be made executable, so that proportion is audited directly. All 1,703 rows parse and 100% of rows agree with the tax-type field, but the executable payload is thin: only 62 of 877 statutory taxation rules carry a statutory rate,

or 7.1%, fifteen carry a formula, and none carries an exclusion condition, so the ceiling is a property of the encoding rather than of the search.

Guided decoding cannot run on CPA items, since it constrains a rate field to a legal enumeration whereas masking option letters is meaningless because every letter is legal; where it does apply it retains 18.6% violations against 6.1% for TaxLogic.

F Complete Results for the 360-Item Controlled Mechanism Attribution

No figure in this table involves any language model call. Its purpose is to test the analytical derivation of Section 4. It does not represent model capability and is never merged with the figures of Table 1. The measured behaviour of the retry family, at 8.6% for $N = 1$ and 8.1% for $N \geq 3$ and never reaching 6.1%, agrees with Proposition 3, in that retries converge to a higher plateau than certification.

Table 2: The rule-derived benchmark, not CPA items. The primary comparison over 360 rule-grounded items, with 120 per domain, in which every retrieval-dependent method shares the same candidate event stream per item. EM denotes exact match, RV the rule violations retained in the delivered answer against ground truth, and Det. RV the violations flagged by the executable checker on delivered answers. Overhead is measured engine time comprising routing, checking, and repair, and is not neural decoding time. Brackets give 95% bootstrap confidence intervals over 2,000 resamples.

Strategy	EM ↑	RV ↓	Det. RV ↓	Overhead (ms)
Vanilla	58.6 (53.3–63.6)	41.4 (36.4–46.7)	24.7	–
Keyword-RAG	78.3 (73.9–82.8)	21.7 (17.2–26.1)	–	6.78
Post-hoc	88.3 (85.0–91.7)	11.7 (8.3–15.0)	2.2	8.94
TaxLogic	79.7 (75.3–83.9)	6.1 (3.9–8.9)	0.0	7.23

G Supplementary Figures

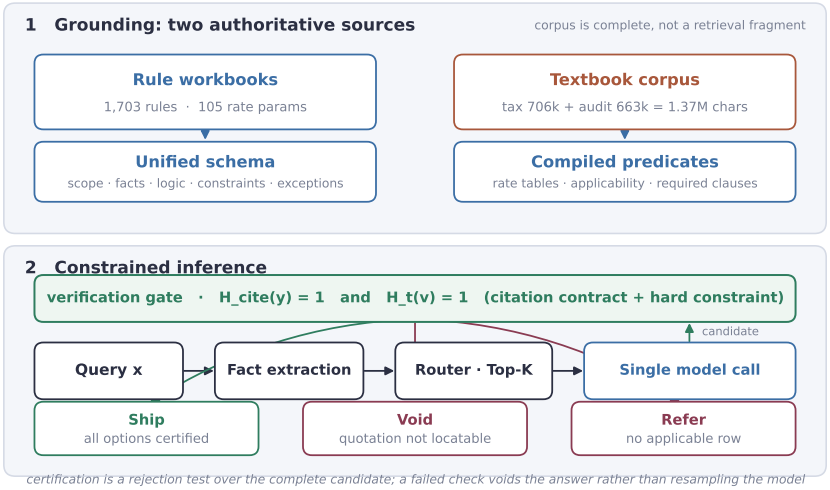

Figure 3: Overview of TaxLogic. A query is subjected to fact extraction, after which a router selects an active set from two corpora, namely codified rules and an authoritative textbook. A candidate answer must satisfy both the hard constraint compiled from the rules and the citation contract, under which every assertion must quote one of the two sources verbatim and must be located against the complete corpus rather than a retrieved fragment.

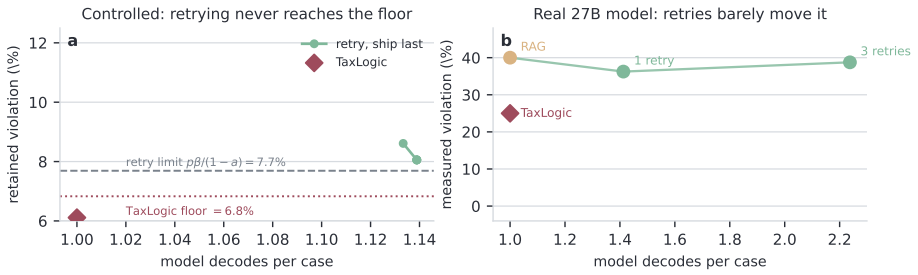

Figure 4: The retry frontier, predicted and measured. (a) Under controlled conditions the retry curve remains above the TaxLogic floor of 6.8% and approaches the retry limit of 7.7%, with model decodes per case on the horizontal axis. (b) On the real 27.9B-parameter model retries barely change the violation rate, which moves from 37.7% to 39.1% and therefore rises slightly.

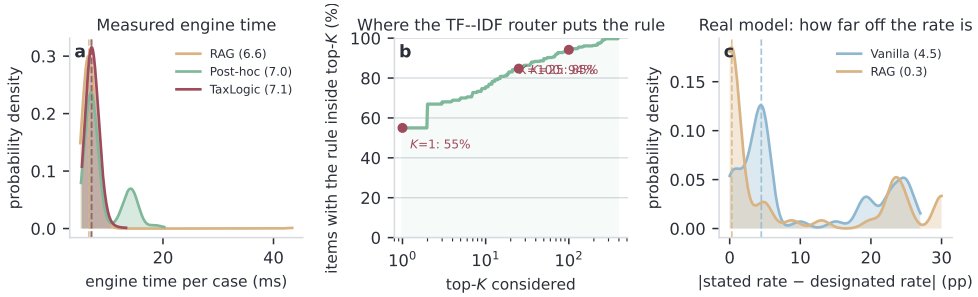

Figure 5: Measured distributions over the 360-item rule-derived benchmark. (a) Kernel density of engine time for three strategies. (b) Cumulative coverage of the governing rule within top K as a function of K. (c) Kernel density of the difference between stated and statutory rate on the real model, comparing Vanilla with RAG.

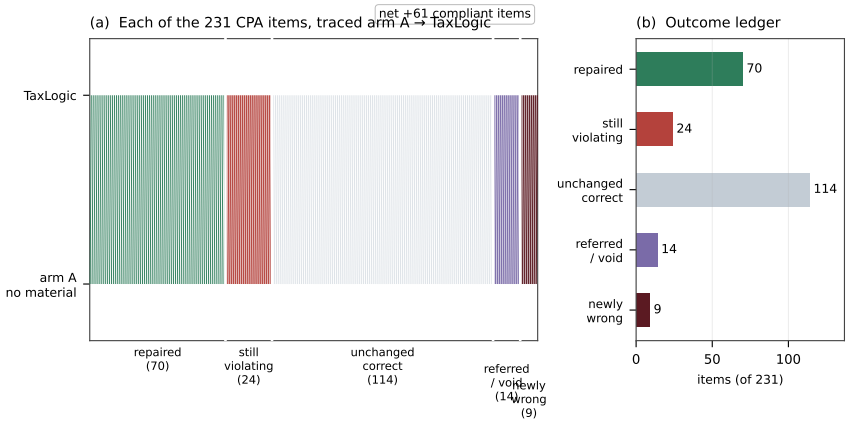

Figure 6: Per-item transition trace over the 231 CPA items. Each segment represents one item, with the outcome of arm A at its lower end and of TaxLogic at its upper end. Five outcome classes are distinguished, namely repaired at 70 items, still violating at 24, already correct at 114, referred or voided at 14, and newly wrong at 9, giving a net gain of 61 compliant items.

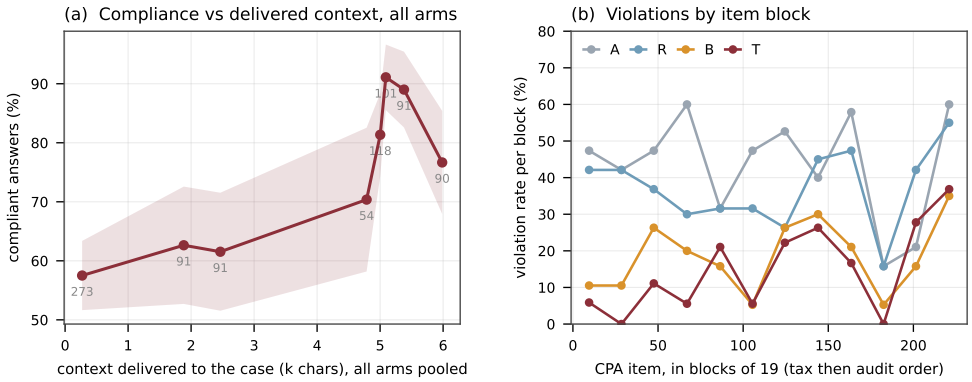

Figure 7: The cost of the contract. (a) Violation rate on the left axis as a solid line and mean model latency in seconds on the right axis as a dashed line, expanded over item type and material condition; latency is essentially level across arms, indicating the safety improvement is close to free in time cost. (b) Binned dose-response curve with the four arms pooled and binned by delivered context size, compliance rising monotonically from 57.5% at 0.3k characters to 91.1% at 5.1k characters, with a binomial 95% confidence band in shading.

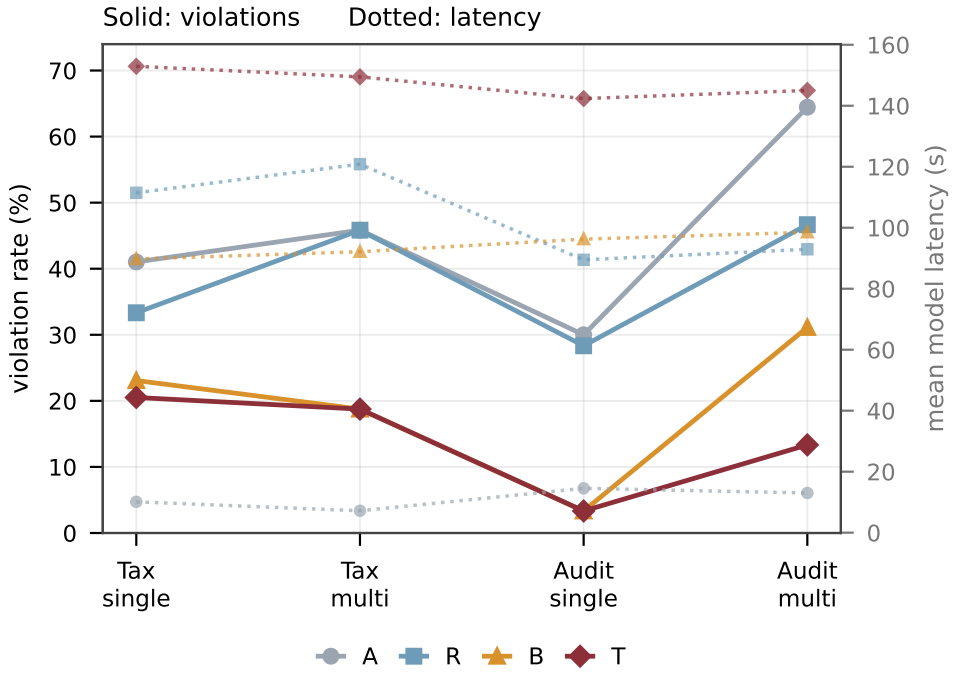

Figure 8: A two-dimensional view of mechanism and cost. (a) Violation rate and mean latency across item types on dual axes. (b) Violation rate by item block of 19 items for the four arms.