INVARIALOOP: OPERATOR-RELATIVE VERIFICATION FOR MULTI-ROUTE INFERENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-route inference systems often map, screen, and combine several candidates before returning one output. Verification can then be mis-scoped: a check on the candidates or an intermediate object is treated as evidence about a different downstream output. We call this *operator-relative verification* and evaluate it by holding candidate realizations fixed while changing only the downstream operation or verifier. Controlled studies show that a mean constraint can hide large violating mass and that heterogeneous structured outputs require canonical mapping before cross-route checks are meaningful. On a public hourly Bike Sharing dataset with 17,379 records, five regressors predict casual rentals, registered rentals, and total rentals under the exact identity `cnt=casual+registered`. Screening reduces the empirical mass that violates the identity from 19.8% to 0.7% in a random split and from 26.0% to 0.5% under a 2011-to-2012 shift; after screening, the mean and violation-mass verifiers make the same decisions. On 100 held-out balance-sheet images from United States Securities and Exchange Commission (SEC) Form 10-Q filings, three vision-language model routes extract Assets and Liabilities-and-Equity while eXtensible Business Reporting Language (XBRL) facts are withheld until scoring. A screened arithmetic average satisfies the accounting identity on all 99 filings it releases but matches both withheld totals on only 56; the intact Qwen3.5-9B constituent matches 94 on those same filings. Among the paired cases, 38 are correct only under Qwen3.5-9B and none are correct only under the average. In a separate study with eight samples per route, with the candidates and arithmetic aggregation fixed, a violation-mass verifier releases one output whose arithmetic mean violates the accounting identity by \$103,878.94, while a mean-residual verifier abstains. These results motivate reporting the candidate set, downstream operator, verified property, final output, and held-out task correctness separately.

1 INTRODUCTION

A *route* is a reproducible candidate-producing configuration, such as a model and processor, one stochastic sample, or an agent with a fixed tool setup. Many inference systems evaluate several routes before returning one result. Agreement can be useful evidence, but shared training data, architectures, prompts, tools, or verifier failures can also produce correlated errors [4, 11, 19, 14].

The evaluation problem becomes sharper when the system transforms the candidate set before release. It may average numerical fields, fuse records, select one route, or normalize units. A check performed before one of these operations need not characterize the object produced afterward.

Concrete failure story. In our study of United States Securities and Exchange Commission (SEC) Form 10-Q filings, two screened vision-language model routes extract Assets and Liabilities-and-Equity from each balance-sheet image, then a downstream rule averages the monetary fields. The average can satisfy Assets = Liabilities-and-Equity exactly while moving away from the withheld reference values. On the 99 filings released by the average, the intact Qwen3.5-9B constituent is correct on both totals in 94 cases and the average in 56; all 38 discordant cases favor the intact constituent. The structural check establishes the accounting relation for the averaged record, not whether either field matches the withheld eXtensible Business Reporting Language (XBRL) reference.

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.

Table 1: Objects distinguished by Invarialoop. The terminology is deliberately ordinary: each row names an object that can differ even when the same upstream candidates are used.

Stage	Object	What a check at this stage can support
Route	Native candidate from one route	Claims about that route output before cross-route mapping
Mapping	Candidate after deterministic mapping	Claims after canonicalization into a shared evaluator space
Screening	Candidates that pass local screening	Claims about which candidates continue to the downstream rule
Combination / selection	Aggregate or selected candidate	Claims about the candidate produced by the downstream operation
Final output	Returned task object	Claims about the task object actually returned to the user or environment

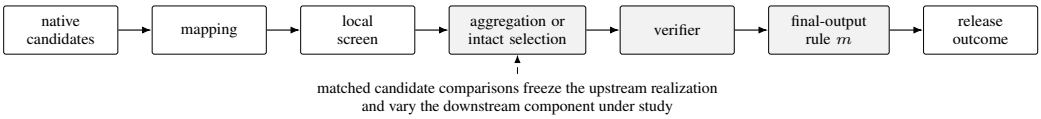

Figure 1: Evaluation path used throughout the paper. Mapping and screening define the admitted candidate set; the downstream operation forms the evaluation object, the verifier checks a declared property, and the final output rule forms the returned task object. The shaded stages are the downstream components varied in matched candidate experiments.

We use *operator-relative verification* for this scope distinction. A result applies to the object and property evaluated under the declared downstream operation. Transfer to a different object requires either a preservation argument or another verification step. Invarialoop records the mapping, local screen, aggregation or intact candidate selection rule, verifier, final output rule, and release outcome, while held-out task correctness remains a separate endpoint.

Our contributions are:

- **Operator-relative verification.** We characterize when a verification result can support a claim after mapping, screening, aggregation or selection, or final output formation, and give counterexamples when the property is not preserved.
- **Matched candidate evaluation.** We give a paired evaluation procedure that holds upstream candidates fixed and changes one downstream operation at a time, separating effects of candidate quality, the downstream operator, and verification.
- **Measured stress tests.** Controlled examples, an hourly bike-rental regression task, typed discrete records, and two SEC 10-Q balance-sheet studies expose different points at which verification scope and held-out correctness diverge.

The contribution is an evaluation method rather than a new abstention theory, ensemble algorithm, or universal verifier. It asks what object was checked, under which transformation, and what empirical claim that evidence supports.

2 Operator-Relative Verification

2.1 Routes, mapping, and screening

Let $R_1, \ldots, R_n$ denote inference routes. A route can include a model, processor, prompt, tool configuration, or other components that causally determine one candidate. Route i produces a native candidate $z_i \in \mathcal{Z}_i$. A deterministic map

$$h_i : \mathcal{Z}_i \to \mathcal{Y}$$

places all candidates in a common evaluator-visible space, yielding $y_i = h_i(z_i)$. This mapping is part of the evaluated system: if two routes encode different semantics under the same coordinate

positions, downstream verification is not meaningful until they are canonicalized. We use I for the admitted route indices, C for the downstream aggregation or selection rule, V for the verifier, m for the final output rule, and o for the returned task object. Other symbols are local to the result in which they appear.

A declared local screen produces $g_i \in \{0, 1\}$ and admitted set

$$I = \{i : g_i = 1\}.$$

Screens may check parse validity, resource limits, local structural properties, or other task-specific conditions. A screen is not assumed to improve task accuracy; it only changes which candidates reach the next stage.

2.2 Aggregation or intact candidate selection

A downstream operator C maps the admitted candidates to an evaluation object,

$$y^C = C(\{y_i : i \in I\}).$$

We use two common classes.

Weighted aggregation. For nonnegative normalized weights γ_i,

$$y^{\text{mix}} = \sum_{i \in I} \gamma_i y_i.$$

Intact-candidate selection. A selector chooses $j \in I$ and returns the unchanged admitted candidate,

$$y^{\text{sel}} = y_j.$$

Formally this is a support-preserving selection operator. The phrase matters only to distinguish it from operators that synthesize a new value.

2.3 Verification, final output, and release

Let V be a declared predicate or statistic applied to y^C. The system may then apply a final output rule m to produce task object $o = m(y^C)$. The release outcome is one of: RELEASE, a non-null output; ABSTAIN, a recoverable null result for which the protocol permits another attempt; or HALT, a terminal null result. A retry is a new attempt and must pass the same declared stages rather than inheriting a previous pass.

Scope rule. A verification result supports the property of the object to which the verifier was applied. If a downstream operation changes that object, the same claim requires either an argument that the operation preserves the property or verification after the operation. Verification of a structural property does not by itself establish task correctness.

2.4 Matched candidate evaluation

Suppose two release rules are compared. If both candidate generation and downstream combination change, an observed difference cannot be attributed to either factor cleanly. Our preferred comparison freezes the upstream candidate realization $\{z_i\}$, mapping $\{h_i\}$, and, when the question concerns the downstream combination or selection rule, the admitted set I, then varies only the intended downstream operator or verifier. This creates a paired experimental unit and prevents evidence from silently migrating between different objects.

Algorithm 1 states the comparison used in the SEC analyses. Held-out task labels are opened only after both arms have serialized their release outcomes.

3 Formal Diagnostics

The results below are deliberately simple. Their role is to falsify common evidence-transfer assumptions rather than introduce new optimization machinery.

Algorithm 1 Matched candidate operator comparison

Require: native candidates $\{z_i\}$, maps $\{h_i\}$, screen g, two downstream arms A, B, held-out labels L

1: Map each candidate once: $y_i \leftarrow h_i(z_i)$
2: Screen once and freeze $I \leftarrow \{i : g(y_i) = 1\}$
3: Record the candidate realization, mapped outputs, and admitted set
4: **for** $a \in \{A, B\}$ **do**
5: Apply only arm a's declared downstream operator, verifier, and final output rule
6: Serialize its release outcome and returned output, if any
7: **end for**
8: Open held-out labels L
9: Compute paired differences on the same experimental units and report release rates separately

Proposition 1: mean agreement does not control violation mass. Let $B = \{y : |y| > \varepsilon\}$ and let $\alpha \in (0, 1)$. For any $c > \varepsilon$, consider a distribution Q that assigns probability $1/2$ to $-c$ and probability $1/2$ to c. It has zero mean but $Q(B) = 1 > \alpha$. Thus a mean criterion and a violation-mass criterion certify different properties.

Proposition 2: weighted-mixture floor under a fixed violation set. For distributions $Q_1, \ldots, Q_n$ and mixture $Q_\gamma = \sum_i \gamma_i Q_i$ with weights $\gamma_i \geq 0$ and $\sum_i \gamma_i = 1$,

$$Q_\gamma(B) = \sum_i \gamma_i Q_i(B) \geq \min_i Q_i(B).$$

If every component exceeds a fixed violation threshold, ordinary mixing cannot make the mixture pass that same threshold. The result does not apply to arbitrary fusion or transformation operators.

Proposition 3: affine structural checks can miss task-relevant drift. Let $r(y) = a^\top y - b$ and $\bar{y} = \sum_i \gamma_i y_i$. Then

$$r(\bar{y}) = \sum_i \gamma_i r(y_i).$$

Hence zero-residual candidates remain zero-residual after averaging, opposing residuals can cancel, and perturbations δ with $a^\top \delta = 0$ are invisible to the verifier. The SEC accounting identity is exactly such an affine check: both extracted totals can drift together while the structural residual remains zero.

4 Evaluation

4.1 Evaluation protocol

The studies increase in realism while preserving a common measurement discipline. Controlled tests create exact counterexamples. An hourly bike-rental regression task adds a public real dataset and temporal shift while retaining an exact algebraic identity. A typed discrete study stresses canonical mapping. The SEC evaluation uses real vision-language-model routes and withholds XBRL values until after the release decision is serialized. We report a fixed primary study with one sample per route and a separate distributional study with eight samples per route; image, prompt, and precision conditions are never pooled into one candidate set.

We report the property used to decide release separately from held-out task correctness. When methods release different numbers of examples, we do not interpret raw accuracy differences as selective-risk dominance unless coverage is matched.

4.2 Controlled falsification

Table 3 summarizes low-compute tests from the executable reproduction suite. The mean-versus-mass battery contains distributions for which a mean check accepts despite large violating mass. Correlated

Table 2: Evaluation matrix. The experiments target different boundaries rather than serving as interchangeable benchmark scores.

Study	Candidate type	Downstream operation	Main quantity separated
Controlled tests	Synthetic samples / sensor estimates	Mean or weighted aggregation	Mean statistic vs. violation mass; correlated error
Bike Sharing	Five regression routes	Uniform average after screening	Screening effect vs. verifier effect under shift
Typed records	Integer triples with different field order	Canonicalize then aggregate	Mapping semantics vs. verification
SEC single-sample	Three structured model records	Arithmetic average of screened fields	Structural validity vs. withheld field accuracy
SEC distributional	Up to 16 screened model samples	Same arithmetic average	Mean-residual vs. violation-mass verifier

Table 3: Controlled falsification results. The right column reports the screened violation-mass rule, not a claim of dominance over other screened rules.

Study	Baseline	Screened violation-mass
Mean vs. mass	75.0% false acceptance (none); 62.5% (mean)	0.0% false acceptance; 75.0% abstain
Sensor walls	44.0% false acceptance	0.0% false acceptance; 76.0% abstain
Sensor obstacles	24.0% false acceptance	0.0% false acceptance; 78.0% abstain
Regression, normal	15.5% violation mass	1.0% violation mass
Regression, correlated	86.5% violation mass	100% abstain

sensor and regression settings show that consensus can fail jointly, while verification-based rules either reject or reduce measured violations depending on whether unsafe routes can be screened.

The method-level tables also show why we do not present the screened violation-mass rule as uniformly superior. In the wall and obstacle studies, a screened mean rule also has 0% false acceptance while abstaining on 44% and 24% of steps, respectively, versus 76% and 78% for the screened violation-mass rule. The 44% and 24% abstention rates exactly match the corresponding unverified-consensus false-acceptance rates because the local step-size screen admits all five routes at every step in these fixed runs; the screened mean is therefore the same mean that unverified consensus releases, but it abstains whenever that mean is invalid. Because coverage differs, these rows diagnose how the verifier changes the decision rather than establish risk–coverage dominance. A post-screening witness shows that the mean/mass distinction is nevertheless real: three admitted residuals $(20, -20, 0)$ have mean zero while two thirds of the admitted mass violates a pointwise tolerance of 10.

4.3 Hourly bike-rental regression with an exact identity

We use all 17,379 hourly records from the Bike Sharing dataset in the University of California, Irvine Machine Learning Repository [6]. Each record contains casual rentals, registered rentals, and their reported total, and the identity `cnt=casual+registered` holds exactly in every row. Five regressors predict the two components and total from calendar and weather features. The label-free local screen is computed on the evaluation batch: each route's mean absolute identity residual is compared with a threshold equal to 0.1 times the training-set mean count. This uses no target labels but is transductive rather than a per-example online screen. We evaluate both a random 75/25 split and a 2011-to-2012 temporal shift.

The exact counts are 860/4,345 violating test predictions before screening and 29/4,345 after screening in-period; under temporal shift they are 2,270/8,734 and 48/8,734. The important measured effect is screening, not marginal superiority of the post-screening mass statistic: after screening, the mean and violation-mass rules coincide. Linear regression and ridge satisfy the identity exactly by linearity on these labels, so part of the screened aggregate's validity is inherited from invariant-consistent generators; the experiment measures whether the release rule isolates that structure rather than claiming to create it.

Table 4: Bike Sharing results at the fixed operating point.

Rule	In-period violation mass	2011→2012 violation mass
Unverified consensus	19.8% (release)	26.0% (release)
Mean-only verification	19.8% (release)	26.0% (release)
Ungated violation-mass	19.8% (abstain)	26.0% (abstain)
Screened mean	0.7% (release)	0.5% (release)
Screened violation-mass	0.7% (release)	0.5% (release)

Table 5: SEC Form 10-Q single-sample study. Structural validity and withheld field accuracy are separate endpoints.

Design	Releases	Structural failures	Both totals correct
Naive consensus	53/100	15/53	7/53
Mean-only verification	38/100	0/38	7/38
Post-hoc violation-mass	38/100	0/38	7/38
Screened mean	99/100	0/99	56/99
Screened violation-mass	99/100	0/99	56/99
Fixed Qwen3.5-9B route	100/100	–	95/100

4.4 TYPED DISCRETE OUTPUTS

Five routes produce integer records (a, b, c) satisfying the intended relation $c = a + b$ but use heterogeneous internal field orderings. A deterministic map canonicalizes each route into a shared schema. Under correlated error, unverified consensus has 86.5% empirical violation mass, while local screening isolates one admitted route and the screened violation-mass rule releases at 5.0%. More importantly, an output-mapping ablation changes normal-condition violation mass from 6.0% with the declared map to 99.5% when incompatible native field orderings are naively combined. Canonicalization is therefore a prerequisite for meaningful cross-route verification, not a cosmetic preprocessing step.

4.5 SEC 10-Q BALANCE-SHEET EXTRACTION: STRUCTURAL VALIDITY VERSUS FIELD ACCURACY

We evaluate balance-sheet extraction from 100 held-out images from SEC Form 10-Q filings. Eligibility requires XBRL facts from the same accession and reporting period for Assets and LiabilitiesAndStockholdersEquity, an XBRL accounting-identity residual at most $1, and a corresponding filed Hypertext Markup Language (HTML) balance-sheet table containing the required total labels [16]. The XBRL values are used only for cohort definition and withheld scoring; they are not available to candidate generation, mapping, screening, aggregation, or the release decision.

The three fixed model and processor routes are Qwen3.5-9B, the bfloat16 checkpoint Ministral-3-8B-Instruct-2512-BF16, and Gemma-4-E4B-it. Each receives the same source image and greedily returns one record in JavaScript Object Notation (JSON) containing reporting period, display units, Assets, and Liabilities-and-Equity. A deterministic map extracts the first complete JSON object, enforces the schema, and normalizes monetary amounts before evaluation. There is no arithmetic repair or reference-assisted parsing.

A route screen is selected using 10 development filings at invalid-rate threshold 0.10: Qwen and Ministral have 0/10 invalid outputs and are admitted; Gemma has 4/10 and is excluded. We use coordinate-wise arithmetic averaging as a simple numeric ensemble operator motivated by standard averaging constructions in ensemble learning [4, 9, 11], not as a claim that financial-document extraction systems customarily average accounting fields. The experiment asks whether a familiar numeric aggregation can preserve the verifier-visible accounting relation while changing field-level task accuracy. The weighted rule averages the two monetary fields across available screened routes. The structural verifier checks the accounting identity at displayed precision before release.

The screened mean and screened violation-mass rules have identical release/abstain outcomes on this one-sample cohort. On the 99 filings released by the arithmetic average, both totals are correct

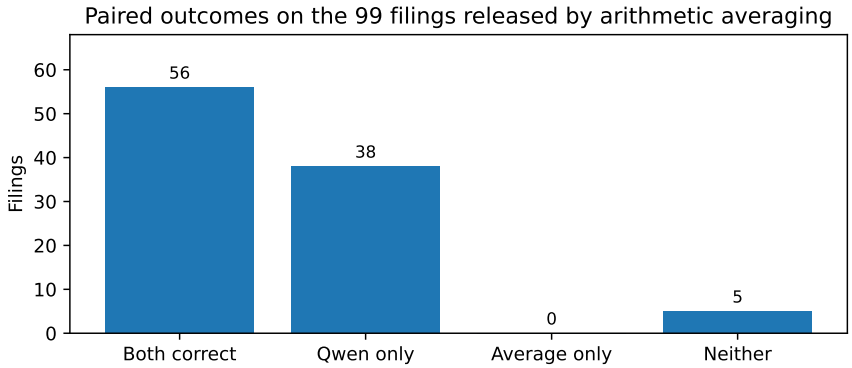

Figure 2: Paired task-correctness outcomes on the 99 filings released by arithmetic averaging. The average is never uniquely correct when the fixed Qwen3.5-9B constituent is wrong; 38 filings are correct only under the intact constituent. This figure concerns task accuracy, not the structural release check.

Table 6: Matched verifier comparison in the SEC study with eight samples per route. Candidates, screening, and arithmetic averaging are held fixed.

Verifier	Releases	Final-output structural failures	Both totals correct
Mean check	93/100	0/93	45/93
Violation-mass check	94/100	1/94	45/94

for the average on 56 and for the intact Qwen3.5-9B constituent on 94: 56 are correct under both, 38 only under Qwen, none only under the average, and five under neither. A paired bootstrap gives an average-minus-Qwen correctness difference of -38.4 percentage points with a 95% confidence interval of $[-48.5, -29.3]$. Across the full 100-filing cohort, Qwen is correct on 95. The comparison is descriptive of this fixed constituent route; it does not show that single-route inference is generally superior or that a learned selector would choose Qwen.

The arithmetic average is genuinely synthetic on a substantial subset: among the 99 releases, the arithmetic mean is identical to an admitted intact candidate on 60 and differs from every admitted candidate on 39. Proposition 3 explains how those synthetic records can remain inside the verifier's affine constraint surface while moving away from the task reference.

On the 38 filings released by both ungated and screened violation-mass rules, both totals match on 7/38 versus 23/38 releases, respectively. This paired common-release comparison is consistent with screening improving the admitted set, but is not a full risk–coverage comparison.

4.6 SEC DISTRIBUTIONAL STUDY: SAME AGGREGATION, DIFFERENT VERIFIER

A second SEC study draws eight samples from each of the same three routes per filing, yielding 24 candidate outputs before development screening. On 10 development filings, Qwen has 3/80 invalid samples, Ministral 0/80, and Gemma 36/80, so Qwen and Ministral are retained and Gemma is excluded. The test-time screened candidate set therefore contains up to 16 candidates per filing. The mean and violation-mass rules use the same screened candidates and the same coordinate-wise arithmetic-mean output; only the verifier changes.

The release decisions differ on exactly one filing. There, 1 of 16 admitted candidates violates the identity, so the violation mass is $1/16 = 6.25\% \leq 10\%$ and the mass rule releases. The arithmetic mean nevertheless has residual of $-\$103{,}878.94$ against a \$1 filing tolerance, so the mean verifier abstains and the output released by the violation-mass rule is structurally invalid. The released output is also incorrect on the withheld totals. This filing gives a direct empirical instance of the boundary between the candidate distribution and the returned arithmetic mean: bounding the fraction

of violating candidates does not, at this threshold, certify that mean. This single disagreement is an existence witness for the failure mode, not an estimate of how frequently it occurs.

5 Related Work

Selective prediction and risk control. Reject-option and selective-prediction methods study when a predictor should abstain [3, 5, 8]; conformal methods provide calibrated coverage or risk guarantees under stated assumptions [17, 2]. Invarialoop does not replace these guarantees. Its focus is whether the object to which a score or guarantee applies is the same object ultimately combined, selected, and released.

Ensembles, routing, and repeated sampling. Mixture-of-experts, Bayesian model averaging, deep ensembles, and self-consistency combine multiple predictive routes [10, 9, 11, 19]. Their main objectives concern prediction, uncertainty, or efficiency. We instead treat aggregation or selection as an evaluation factor whose output may require its own evidence.

Runtime verification and agentic refinement. Runtime monitoring and shielding enforce task-specific properties at execution time [13, 1]. Modern agents also use execution feedback and self-correction [20, 15]. Invarialoop is complementary: a new candidate generated after feedback is a new attempt and does not inherit a previous pass.

Structured decoding. Grammar- or schema-constrained decoding can guarantee syntactic properties of generated candidates [7, 18], but it changes the candidate-generation distribution rather than certifying semantic or task-level correctness. In our terminology, decoding constraints belong upstream of the release verifier: a syntax-valid candidate must still pass the same mapping, screening, and task-specific checks as a freely decoded candidate.

The software-security benchmark SEC-bench evaluates intact patches and proof-of-concept artifacts in reproducible environments [12]. Such settings illustrate why the downstream operation must be named explicitly: averaging patches is generally meaningless, whereas selecting one intact candidate and then executing the benchmark verifier is well defined.

6 Discussion

The experiments do not identify one universally preferred release rule. They show that the evidence available for a claim depends on the object that reaches the verifier. When a downstream operation is known to preserve the property, that preservation can justify transfer. Otherwise the downstream object must be checked directly. Task accuracy remains a separate endpoint whenever the operation changes the task object, as it does when monetary fields are averaged or an intact software patch is selected.

In the single-sample SEC study, arithmetic aggregation preserves the accounting identity on 99 releases while reducing held-out field accuracy relative to one fixed constituent on the same filings. The comparison does not optimize a selector and the routes need not have equal cost or coverage. The distributional study exposes a different boundary: a candidate-level violation-mass condition passes even though the arithmetic mean returned downstream violates the accounting identity. These are distinct failures, which is why the downstream operator and verifier should be varied separately.

Structured decoding fits the same account. A grammar can guarantee membership in a syntax-defined language, while semantic checks still occur downstream. Some artifacts, including software patches and tool plans, are naturally evaluated intact rather than averaged. The verifier therefore depends on both the candidate representation and the operation applied before release.

7 Limitations

Verifier quality is external. Invarialoop cannot make an incorrect or incomplete check sound. The SEC study intentionally demonstrates this limitation: satisfying the accounting identity does not imply extracting the correct totals.

Screening changes the estimator. Local screening can discard useful information and alter statistical properties of a pooled estimator, including bias and variance. Invarialoop does not imply that stricter screening improves predictive accuracy or estimator quality unless those properties are themselves part of the declared evaluation objective. The framework identifies which candidates and transformations produced a released object; it does not make screening universally beneficial. The Bike experiment additionally uses a label-free but transductive batch screen, so its measured benefit should not be read as a result for per-example online gating.

Auditability is not deployment safety. Reconstructing why a rule released an output does not establish population reliability, calibration, or downstream safety.

Finite-sample checks have finite-sample scope. When a rule operates on an empirical candidate distribution, the result establishes conformance on that finite decision sample unless an additional calibration or concentration argument supports population generalization.

Combination and selection rules have different attainable sets. The mixture-floor result applies only to weighted mixtures under a fixed violation set. Typed fusion, learned transformation, and intact selection require separate analysis.

The SEC cohort is conditioned. The SEC claims apply to the eligible filings in this evaluation and are not a general document-understanding benchmark. We use this setting because it cleanly separates a structural invariant checked before release from machine-readable withheld field truth.

The paper studies release, not persistent learning. Persistent memory, skill mutation, adapter updates, and model-weight updates are outside the evaluated release object.

8 Conclusion

Invarialoop treats the transformations between candidate generation and release as part of the evaluation rather than as bookkeeping. The experiments show why: screening can change which evidence reaches an aggregate, canonical mapping can determine whether a structural check is meaningful, arithmetic aggregation can preserve an invariant while changing field accuracy, and a candidate-level mass condition can fail to certify the downstream arithmetic mean. Claims about a released object therefore require either verification of that object or an argument that the intervening transformation preserves the verified property. Matched candidate comparisons make this testable by holding upstream outputs fixed while changing the downstream operator or verifier on the same experimental units.

Reproducibility Statement

The anonymous supplementary material contains code for the controlled stress tests and Bike Sharing experiment, together with per-filing tables and analysis code for the SEC studies. The appendix lists the main thresholds, seeds, model routes, and data-access steps. The controlled and SEC analyses can be recomputed without model inference; the Bike script retrieves the public dataset before running the reported conditions.

Artificial Intelligence Use Statement

Generative artificial intelligence (AI) tools were used to assist with literature search and summarization; refinement of the conceptual framing and hypotheses; formulation and checking of mathematical statements; feedback on experimental design; drafting and refactoring analysis code; interpretation checks; figure preparation; and manuscript drafting and editing. Controlled synthetic examples and some analysis code were developed with generative artificial intelligence assistance. Reported numerical results are tied to the submitted scripts or per-example tables, and AI-assisted mathematical and empirical claims were reviewed against the corresponding derivations or artifacts before submission. The authors take responsibility for the final content.

REFERENCES

[1] Mohammed Alshiekh, Roderick Bloem, Ruediger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. Safe reinforcement learning via shielding. In *AAAI Conference on Artificial Intelligence*, 2018.

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

[3] C. K. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970.

[4] Thomas G. Dietterich. Ensemble methods in machine learning. In *Multiple Classifier Systems*, pp. 1–15, 2000.

[5] Ran El-Yaniv and Yair Wiener. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11:1605–1641, 2010.

[6] Hadi Fanaee-T. Bike sharing. UCI Machine Learning Repository, 2013.

[7] Saibo Geng, Martin Josifoski, Maxime Peyrard, and Robert West. Grammar-constrained decoding for structured NLP tasks without finetuning. In *EMNLP*, 2023.

[8] Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*, 2017.

[9] Jennifer A. Hoeting, David Madigan, Adrian E. Raftery, and Chris T. Volinsky. Bayesian model averaging: A tutorial. *Statistical Science*, 14(4):382–401, 1999.

[10] Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991.

[11] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, 2017.

[12] Hwiwon Lee, Ziqi Zhang, Hanxiao Lu, and Lingming Zhang. SEC-bench: Automated Benchmarking of LLM Agents on Real-World Software Security Tasks. In *Advances in Neural Information Processing Systems*, 2025.

[13] Anna Lukina, Christian Schilling, and Thomas A. Henzinger. Into the unknown: Active monitoring of neural networks. In *Runtime Verification*, pp. 42–61, 2021.

[14] Potsawee Manakul, Adian Liusie, and Mark Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *EMNLP*, 2023.

[15] Jie Ren, Sercan O. Arik, Xuefeng Chen, Jilin Chen, Chao Yang, Rui Sun, and Jinsung Yoon. Sets: Leveraging self-verification and self-correction for improved test-time scaling. *Transactions on Machine Learning Research*, 2025.

[16] U.S. Securities and Exchange Commission Staff. Edgar xbrl guide, 2026. June 29, 2026.

[17] Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, 2005.

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

[19] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023.

[20] John Yang, Carlos E. Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. Swe-agent: Agent-computer interfaces enable automated software engineering. In *Advances in Neural Information Processing Systems*, 2024.

A Reproducibility Details

The anonymous supplementary code contains three executable components aligned with the manuscript terminology: controlled stress tests, the Bike Sharing experiment, and analysis of the SEC per-filing outputs. The controlled tests and SEC analyses run on a central processing unit (CPU) only. The Bike script downloads the public Bike Sharing data and reproduces both the random-split and temporal-shift conditions. The SEC analysis package contains the per-filing mapped outputs and held-out evaluation fields needed to reproduce Tables 5–6 and Figure 2; it does not require model inference.

Controlled-test parameters. All controlled studies use fixed random seeds. The mean-versus-mass battery uses mean tolerance 0.5 and violation-mass threshold 0.10. The sensor studies use five routes, 200 samples per route per step, 50 steps, and mass threshold 0.05. The algebraic regression study uses five regressors, 500 training examples, 200 test examples, pointwise residual tolerance 0.5, and mass threshold 0.10. The structured-record study uses 200 integer records per condition and canonicalizes every route to (a, b, c) before checking $c = a + b$.

Bike Sharing parameters. The five regressors are linear regression, ridge regression ($\alpha = 1$), a decision tree (depth 5), random forest (50 trees, depth 5), and gradient boosting (50 estimators, depth 3), all with fixed seed 42 where applicable. Features are season, month, hour, holiday, weekday, working-day status, weather category, temperature, apparent temperature, humidity, and wind speed. The identity tolerance is 0.1 times the training-set mean total-rental count and the violation-mass threshold is 0.10. The in-period condition uses a seed-42 random 75/25 split; the shift condition trains on 2011 and evaluates on 2012.

SEC evaluation fields. For each filing, the supplementary tables provide the deterministic mapped fields used by the release rules, structural verification outcomes, release outcomes, and withheld XBRL-based task scores. The single-sample study uses one output from each of the three named routes. The distributional study uses eight outputs per route and applies the screening decision selected on development filings before comparing the two verifiers. All comparisons reported in the main text are recomputed from these per-filing tables.

B Additional Diagnostics

Attribution stability. A controlled leave-one-out diagnostic evaluates whether route-responsibility rankings remain stable as the number of generators grows. Mean Spearman correlation between bootstrap and reference rankings is 0.9726 with three generators, 0.9189 with five, 0.9424 with eight, 0.8977 with twelve, 0.8051 with sixteen, and 0.6332 with twenty. Because the ranking becomes tie-heavy and unstable at larger route counts, we treat fine-grained attribution as a diagnostic rather than a core contribution.

Release outcomes across retries. Two controlled traces illustrate the distinction between recoverable abstention and terminal failure. In one condition, empirical violation mass is 25.0% for the first two attempts and falls to 1.5% on the third, yielding ABSTAIN, ABSTAIN, RELEASE. In a second condition, five locally admitted routes share the same 25% violation pattern and repeated attempts do not improve the constructed output, so the procedure eventually returns HALT. These traces illustrate release-state semantics; they are not evidence that the feedback update is an optimal learning rule.

C Reference-Value Separation in the SEC Studies

XBRL facts are used for cohort definition and post-decision scoring. They are not exposed to the model prompts, deterministic mapping, local screening, arithmetic aggregation, or verifier. The per-filing supplementary tables therefore support two distinct evaluations: whether the returned record satisfies the declared structural check, and whether its fields match the withheld reference values. The main paper reports these endpoints separately throughout.