LOCI

WHITE PAPER · V1 · JULY 2026

# The Discovery Bottleneck

Fraud defense got fast at deploying detection logic. It never got fast at discovering it. Why discovery is now the binding constraint on adaptation, and the ensemble that credible automated discovery requires.

A COMPANION TO "THE FRAUD LANGUAGE MODEL (FLM)", JANUARY 2026

EXECUTIVE SUMMARY

# The queue moved upstream

For most of the history of fraud defense, the constraint was engineering. Shipping a new rule, model, or feature meant development queues, release cycles, and integration projects. Weeks passed between insight and protection. That constraint has largely dissolved. Modern ML infrastructure, real-time feature stores, streaming platforms, and structured rule runtimes have made deployment fast: once a team knows what to detect, expressing and shipping it is measured in hours.

> **THE CORE CLAIM**
>
> The core challenge has shifted from expression to discovery. The question is no longer how to write and deploy detection logic. It is how to know what needs to be written in the first place.

Fraud teams drown in device, behavioral, and transactional data while the malicious signal stays a needle in a growing haystack. Analysts spend their days reacting to alert queues instead of hunting the coordinated fraud rings beneath them. Supervised models are structurally blind to novel patterns until discovery hands them labels to learn from. Discovery remains where deployment was fifteen years ago: artisanal, slow, dependent on individual analysts, and invisible to governance. It is now the binding constraint on how fast an institution adapts to fraud, and through the cold-start dependency, on how fast everything downstream of it learns.

This paper examines why discovery has resisted automation. The obvious answers (black-box models, generic AI rule-writing, unsupervised data mining) each fail in a characteristic way. It then presents the **ensemble of techniques** we found necessary to automate discovery credibly: interpretable-by-construction mining, time-honest validation, statistical evidence grading, generation mechanically verified against evidence, outcome reconciliation that respects label latency, selection under analyst-capacity budgets, compounding pattern memory, and governance enforced in architecture rather than policy. No single technique closes the gap. The ensemble does.

These lessons are not theoretical. They come from building **Autographer**, our fraud-pattern discovery engine, and from the failures we hit along the way, several of which we document candidly here. But the argument stands apart from any product: institutions that industrialize discovery, under evidence a regulator can hold, will adapt at a speed that manual pattern-hunting cannot match.

CONTENTS

# What's inside

CHAPTER 01

# The Great Inversion

## When deployment was the bottleneck

A decade ago, the fraud team that knew what to detect still waited on engineering to detect it. Rules lived in code. Models lived in quarterly release trains. A new feature meant a data-engineering project. The industry responded, and the response worked: real-time feature stores serve fresh aggregates in milliseconds, model platforms retrain and ship continuously, and structured rule runtimes (FLM-class systems among them) compile analyst intent into deterministic, auditable execution within minutes. Expression is solved infrastructure.

## The queue moved upstream

Deployment speed did not eliminate the delay between attack and defense. It relocated it. The weeks now sit before the first line of logic is written.

**Data deluge, insight drought.** Fraud teams are drowning in device, behavioral, and transactional data, and every new channel or signal provider deepens the pile. The malicious signal is in there somewhere. Finding it by hand is a needle-in-a-haystack search where the haystack grows faster than any team can staff.

**Discovery is artisanal.** Patterns are found by analysts with SQL, dashboards, and intuition. The craft is real, but it scales with headcount, walks out the door with attrition, and cannot run overnight.

**Analysts are trapped in the reactive loop.** Review queues are flooded. The working day goes to triaging alerts from existing rules, which leaves no capacity for the proactive work that matters most: hunting the underlying, coordinated fraud rings whose patterns no current rule describes. The team that should be discovering is fully booked reacting.

**Discovery is unbudgeted.** Every fraud team keeps an informal backlog of "we should write a rule for that." Attack campaigns end before their rules ship, not because deployment is slow but because discovery never started.

**Discovery is invisible to governance.** When an examiner asks why a rule exists and what evidence supports it, the discovery work that justified the rule usually lives in a former analyst's memory and an abandoned notebook. The most consequential step in the control lifecycle leaves the thinnest artifact.

**Discovery decays silently.** The mirror image of finding new patterns is noticing that old ones stopped working. Almost no institution re-measures deployed rules against outcomes systematically, so alert queues fill with the residue of rules nobody re-validated.

Meanwhile the adversary industrialized. Fraud-as-a-service operations use AI to mutate tactics continuously, and the attacker's discovery loop (probe, observe, adapt) runs at machine speed. A 2026 Tsinghua-led field study of e-commerce refund fraud documents generative AI fabricating hyper-realistic defect evidence and synthetic personas across four distinct threat vectors, with defenders reporting that verification "increasingly fails" against advancing capability while attack costs approach zero (Zhang et

al., arXiv:2606.03215). Every such GenAI-enabled vector is, by construction, a pattern with no historical labels: cold-start fraud, invisible to trained models until someone discovers it. A defense whose discovery loop runs at analyst speed loses on cadence no matter how fast its deployment pipeline is.

## The cold-start dependency

One structural fact elevates discovery from a bottleneck to *the* bottleneck: machine learning models cannot see what they were never taught. Supervised models learn from historical labels. When a genuinely new fraud pattern emerges there are no labels, and the model stays blind until the pattern is discovered, characterized, and fed back as training signal. This is the cold-start problem. It means discovery is not an alternative to the ML stack; it is the ML stack's upstream dependency. Every detection technology an institution operates, whether rules, models, or ensembles, learns what to look for from the same starved source. Accelerate discovery and everything downstream of it gets smarter sooner.

---

**The industry spent a decade making the last mile fast**
**while the first mile stayed on foot.**

---

CHAPTER 02

# Why Discovery Resists Automation

If discovery is the bottleneck, why hasn't it simply been automated? Because the three obvious approaches each fail a requirement that regulated fraud defense cannot waive.

**Black-box models "discover," but cannot hand over the pattern, and cannot start cold.** A gradient-boosted ensemble absolutely encodes discovered structure, but as a decision boundary no one can read. The discovery is real yet inseparable from the model. It cannot be extracted as a rule an analyst can review, a committee can challenge, or a runtime can execute independently. Post-hoc explanation approximates the model rather than revealing it, and the fidelity of that approximation is itself contestable under examination. (Our FLM paper treats this critique in depth; it carries over unchanged.) The cold-start dependency of Chapter 1 also bounds what any supervised model can contribute here. Against a genuinely novel pattern there are no labels to have learned from, so the model is just as blind as the rule set until discovery gives both something to see.

**Generic AI rule-writing is fluent, not faithful.** Large language models translate data summaries into plausible rule logic, and unchecked, they invent logic the data does not support. Building a discovery system, we catalogued the failure modes first-hand: generated rules referencing fields that did not exist in the data vocabulary; correct fields with prose-styled values that could never match the normalized data; a boolean condition serialized as text, syntactically present but matching zero records; and statistically strong conditions simply dropped from the written rule. Each of these failures looks like a working rule. That is what makes them dangerous.

**Unsupervised mining drowns the analyst it was meant to save.** Classical pattern mining over high-dimensional fraud data yields thousands of correlations, most of them spurious artifacts of multiple comparisons over non-stationary data. Without statistical discipline (out-of-sample validation, evidence grading, honest handling of sparse counts) automated mining converts an analyst shortage into an alert-triage crisis one step earlier in the pipeline.

> THE REQUIREMENT
>
> Automated discovery is credible only if the system can discover patterns statistically, express them symbolically, prove the expression matches the evidence, quantify that evidence honestly, and place a human in structural control of what ships. No single technique delivers all five. The next chapter describes the ensemble that does.

CHAPTER 03

# The Ensemble: What Credible Automated Discovery Requires

Eight techniques close the gap together; none is sufficient alone. For each we state the requirement, the mechanism that satisfies it, and where we earned it, the scar that taught us.

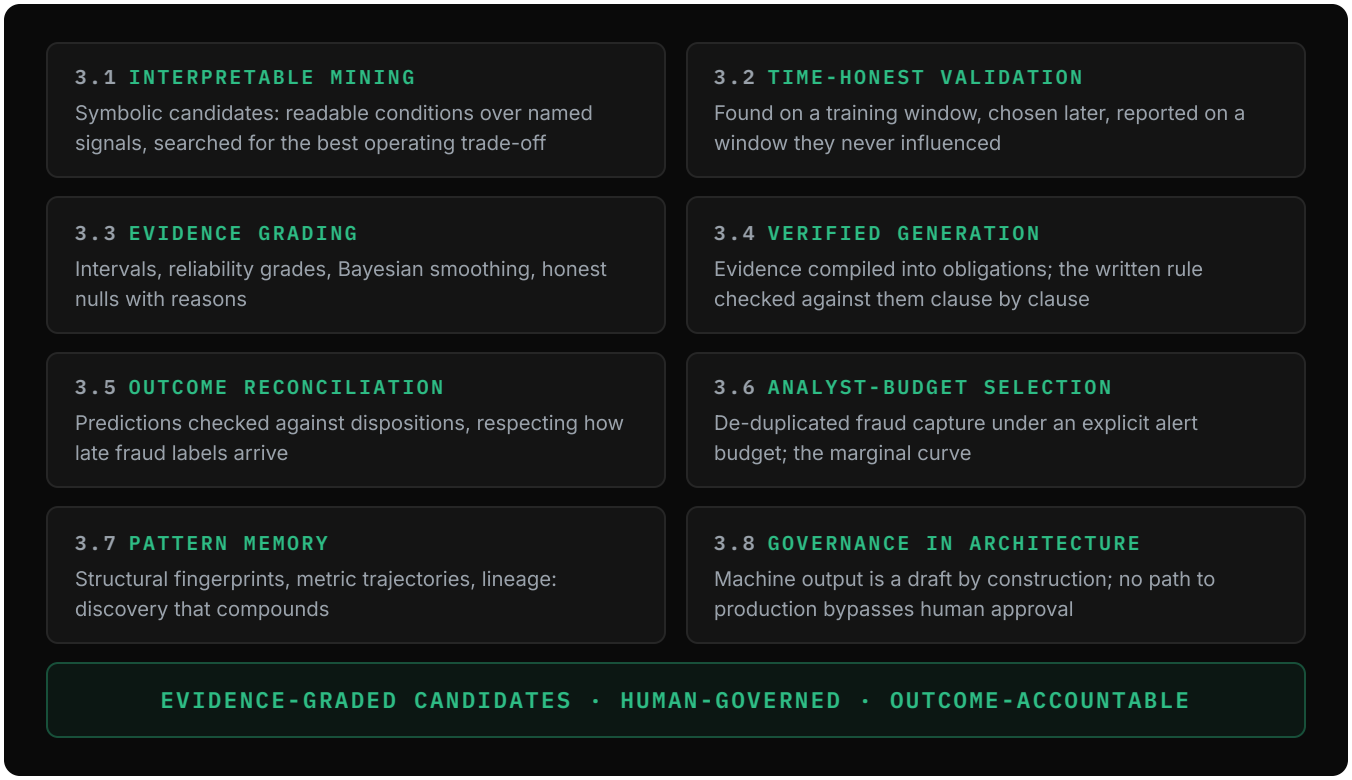


Fig. 1 · The ensemble: each technique answers a failure mode that does not go away

## 3.1 Mine interpretable structure, or the output can't be governed

Discovery must produce symbolic candidates, meaning readable conditions over named signals, because everything downstream (verification, review, challenge, audit) depends on the artifact being inspectable. This constrains the mining layer to strategies whose outputs are conditions rather than weights: single-signal thresholds, cross-signal conjunctions that combine different signal families (amount with geography with beneficiary, rather than three variants of one idea), categorical evidence scoring, and temporal or velocity patterns. Searching thresholds directly for the best precision/recall trade-off under an alert-rate ceiling, instead of accepting fixed percentiles, matters more than it sounds. Fixed cutoffs routinely surface a weak variant of a signal whose strong form sits one search away.

## 3.2 Validate forward in time, or the metrics flatter you

Fraud data is non-stationary by adversarial design. Random cross-validation leaks the future into the past and systematically inflates results. The honest protocol is sequential: patterns are found on a training window, chosen on a later selection window, and reported on a still-later window they never influenced. Time-ordering must survive the entire pipeline, including any sampling of large inputs, which must be uniform (stratification silently changes the base rate every downstream metric assumes) and deterministic (a re-run of the same job must see the same data). This is the difference between metrics that predict production behavior and metrics that flatter a demo.

## 3.3 Grade the evidence, don't just report it

A precision of 0.91 on six cases and on six hundred are different claims, and a system that prints both as "0.91" teaches reviewers to distrust everything it prints. Every surfaced metric needs its statistical context attached: confidence intervals; a reliability grade (strong, limited, weak, or insufficient) derived from sample and positive counts; Bayesian smoothing for categorical evidence, so a value seen six times cannot be scored like one seen six hundred times; and honest nulls. Where a statistic cannot be soundly computed, such as lift on a window with too few confirmed cases, report null with a reason rather than a numeric artifact.

| EVIDENCE | NAIVE PANEL | EVIDENCE-GRADED |
|---|---|---|
| **600 cases, 546 confirmed** | 0.91 | `0.91 [0.88, 0.93] · STRONG` |
| **11 cases, 10 confirmed** | 0.91 | `0.91 [0.62, 0.98] · WEAK` |
| **Sparse window, 4 cases** | 0.25 | `null · INSUFFICIENT, with reason` |

Fig. 2 · The same headline number, different claims. Try it live: runloci.com/fraud-ai-discovery-bottleneck

> **SCAR TISSUE**
>
> We learned this one the hard way. An early version reported lift as zero on sparse windows, and that single misleading zero, sitting beside legitimate numbers, undermined the credibility of the whole panel.

## 3.4 Ground the generation, then verify it mechanically

The step from validated pattern to written rule is where AI-assisted authoring fails silently. This is the failure catalogue of Chapter 2, and the countermeasure has two halves. **Grounding:** compile the evidence into exact machine-readable constraints (field, operator, value, each with its support, precision, and recall) and hand those to the generator as obligations rather than inspiration, alongside a manifest of what the target runtime can execute. **Verification:** compare the generated rule against those constraints clause by clause, with values normalized to the data's canonical forms, dropped high-evidence conditions flagged,

distorted or dead clauses flagged, and aggregations checked for correct entity scoping. Verification warns rather than blocks, since a rule that omits a condition may still be the right rule, but the omission is never silent.

> THE PRINCIPLE
>
> Fidelity between evidence and artifact is measured by the system, not asserted by it. The language model is confined to explanation and translation, invoked only after all statistics are final. Remove it entirely and the same patterns emerge with the same evidence.

### 3.5 Reconcile predictions against outcomes, respecting label latency

Discovery-time metrics are predictions, and a discovery system that never checks its predictions is an unaccountable oracle. The closing mechanism is an outcome contract. The institution reports back, per running candidate, aggregate windows of alerts fired, confirmed fraud, confirmed legitimate, and pending review, and the system reconciles observed precision and alert rate against its held-out predictions, with intervals. The subtlety that decides whether this works: fraud confirms late. Early windows are dominated by pending dispositions, so naive observed precision is biased low and would indict good rules during the confirmation lag. Drift may be flagged only when a window is statistically mature, meaning sufficiently dispositioned, and the observed interval excludes the prediction. A drift detector that punishes rules for slow labels teaches operators to ignore drift detection, which is worse than not measuring at all.

### 3.6 Optimize for the analyst budget, not the leaderboard

The operational question about a candidate rule set is not its F1 score. It is how many alerts per week the team will face, and how much fraud gets caught for that workload. Selection must therefore optimize de-duplicated fraud capture under an explicit alert budget, expressed as volume or as analyst-hours through a handling-time estimate. "De-duplicated" is load-bearing here: two rules firing on the same event cost one investigation, so the selection mathematics must count set unions, never sums. The most useful artifact this produces is the marginal curve, each additional rule's incremental fraud against its incremental alerts, which is the trade-off fraud-operations leadership actually reasons about when sizing a team.

### 3.7 Remember across runs, or discovery never compounds

One-shot mining rediscovers the same patterns forever and notices decay never. Give every pattern a structural fingerprint, an identity stable across runs even as thresholds move, and accumulate history against it: recurrence across generations, metric trajectories, observed threshold drift, lineage, and real-world outcome records. Discovery then compounds. Recurring strong patterns are recognized and deepened rather than re-derived. Decaying ones show falling trajectories instead of vanishing silently. The institution can answer, per pattern, what changed between generations, which is simultaneously the tuning conversation and the audit trail. We attach the word "generational" to this observable lineage across generations of runs, and we treat "quality improves generation over generation" as an empirical claim to be measured, not assumed.

### 3.8 Enforce governance in architecture, not policy

Every prior technique is undone if automation can quietly ship logic. The guarantees must be structural. Machine output is a draft by construction, in a lifecycle whose state transitions are enforced by the system itself; there is no path from generated to deployed that bypasses human approval. Deployment is executed by the institution's own engine, outside the discovery system entirely. Raw input data is processed in memory and never durably stored, and all artifacts are isolated per organization. When these are properties of the architecture rather than promises in a policy document, the compliance conversation changes character. The reviewer is no longer trusting a workflow. They are inspecting a mechanism.

CHAPTER 04

# The Outcome: Discovery as an Institutional Capability

Assemble the ensemble and the character of fraud defense changes.

**The discovery loop matches the deployment loop.** Going from "pattern exists in the data" to "evidence-graded candidate awaiting human review" becomes a pipeline run plus a review session. The institution's adaptation cadence is set by how often it runs discovery and how fast humans judge, not by how fast humans can dig.

**Analysts move up the stack, from reacting to hunting.** The scarce resource is experienced fraud judgment. It stops being spent on excavation and alert triage and goes to evaluation and pursuit: reviewing candidates that arrive with their evidence, intervals, warnings, and lineage attached, and following the threads automated discovery surfaces toward the coordinated fraud rings that reactive queue-work never reaches.

**The cold-start gap narrows for the whole stack.** Because discovery characterizes novel patterns early, and outcome reconciliation confirms them, it manufactures the very thing supervised models starve without: labeled, validated examples of the new attack. Industrialized discovery is how the institution's models, not just its rules, learn about a pattern weeks sooner.

**Governance gets an artifact instead of an anecdote.** Why does this rule exist? The candidate itself answers: the mined evidence, its out-of-sample provenance, the verification findings, the approval trail, and once deployed, its predicted-versus-observed record.

---

**Exam preparation**
**becomes retrieval.**

---

**The rule portfolio becomes maintainable.** Outcome reconciliation and pattern history give standing evidence for the work that never gets staffed: retiring decayed rules, consolidating overlapping ones, and defending alert budgets with marginal curves rather than intuition.

We built this ensemble as **Autographer**, our fraud-pattern discovery engine. It is the system in which every mechanism above is implemented and from which every disclosed failure was learned. It emits candidates as FLM specifications, completing the pipeline our FLM paper began: one system discovers and writes, one executes, and humans govern both. The thesis of this paper is larger than any product, though. The ensemble is what any credible industrialization of fraud discovery will need, because each element answers a failure mode that does not go away.

CHAPTER 05

# Regulatory Alignment

The transparency analysis from our FLM paper carries over intact. Automated discovery strengthens it at one specific point: the evidence behind each control.

EUROPEAN UNION

**GDPR Art. 22 · AI Act**

"Meaningful information about the logic involved" is satisfiable when logic is a readable rule, and more defensible when the rule arrives with the statistical evidence that justified it.

UNITED STATES

**SR 11-7 Effective Challenge**

The ensemble produces the challenge package by default: evidence with intervals and provenance, verification findings, and the full review trail.

AFRICA

**Emerging Frameworks**

Automating discovery while holding approval human serves the guidance where analyst capacity is scarcest: its letter and its operational intent.

**European Union.** GDPR Article 22's "meaningful information about the logic involved" and the AI Act's transparency posture are satisfiable when decision logic is a readable rule, and more defensible still when the rule arrives with the statistical evidence that justified it, validated on the institution's own data.

**United States.** SR 11-7's "effective challenge" presumes logic and evidence that subject-matter experts can interrogate. A discovery pipeline built on the ensemble produces the challenge package by default: evidence with intervals and provenance, verification findings, and the full review trail.

**Africa and emerging frameworks.** Central-bank guidance converging on explainability meets institutions where analyst capacity is scarcest. Automating discovery while holding approval human serves both the letter of the guidance and its operational intent.

THE TRAJECTORY

Regulators increasingly ask not only what a control does but why it exists and what supports it. Institutions whose discovery process yields durable evidence will answer with an artifact. The rest will answer with archaeology.

CHAPTER 06

# Looking Ahead

**Published, reproducible evaluation.** This paper reports no detection-performance figures, deliberately. Claims about discovery quality deserve the same discipline the ensemble imposes internally, so our benchmark program publishes its design first and ships every result with data, seeds, harness, and failure analysis. It measures generation fidelity executably (does the written rule reproduce its evidence when run against the data), audits the verification layer's own accuracy, tests deployment fidelity into target engines, and reports detection efficiency against tuned gradient-boosted baselines as the false-positive rate at fixed recall, the operating trade-off, rather than a leaderboard number. Its first artifact is already built: a deterministic synthetic benchmark with planted ground-truth rules, so recovery and fidelity are measured against an answer key rather than labels alone.

**Calibration.** As outcome data accumulates, labeled heuristic confidence can graduate to calibrated probability, fitted per institution on real dispositions and only where data suffices. A system that refuses to print a probability it cannot back today earns the right to print one tomorrow.

**The adversarial trajectory.** Transparent rules invite threshold-gaming. The countermeasure is not opacity but cadence: a continuously re-evidenced portfolio whose drift surfaces in outcome reconciliation. Deliberate adversarial hardening (randomized deployment, decoy thresholds) is an open research direction we approach without overclaiming.

**The hybrid horizon.** Learned feature extractors emitting scores that symbolic rules consume as ordinary conditions would let perceptual fraud signals feed auditable logic. That is the principled path past the symbolic ceiling without surrendering decision transparency.

CHAPTER 07

# Conclusion

The industry solved the problem it could see: deployment. Feature stores, model platforms, and structured rule runtimes made expressing and shipping detection logic fast, and in doing so exposed the problem that was always underneath. Knowing what to write. Discovery is now the constraint, and it cannot be closed by hiring alone, by black boxes that won't hand over the pattern, or by generative fluency without verification.

It can be closed by an ensemble: interpretable mining, time-honest validation, graded evidence, mechanically verified generation, outcome reconciliation that respects how fraud labels actually arrive, selection under real analyst budgets, memory that compounds across runs, and governance enforced in architecture. Each element answers a specific failure mode. Together they make discovery something an institution operates, rather than something it hopes its best analyst does before leaving.

Fraud defense spent a decade making the last mile fast. The next advantage belongs to whoever industrializes the first mile, with evidence a regulator can hold and a human hand on every deployment.

---

**The future of fraud defense is explainable.**
**It begins with discovery.**

---

ABOUT LOCI

## Fraud control that shows its work

Loci Fraud AI builds intelligent fraud defense infrastructure for financial institutions across Africa, Europe, and North America. Our platform, including transaction monitoring, AccessGate continuous authentication and biometrics, and Autographer AI fraud-pattern discovery, the reference implementation of the ensemble described in this paper, helps banks and fintechs detect fraud in real time while maintaining full regulatory compliance.

Founded by practitioners with deep experience in scalable technology and AI, Loci combines rigorous engineering with operational pragmatism. We believe fraud defense should be powerful *and* understandable.

WHY NOW

Attackers iterate with AI in hours; manual pattern discovery answers in weeks. Deployment is no longer the bottleneck. Discovery is. Loci gives fraud teams the discovery engine to keep pace: every candidate evidence-graded, human-approved, and ready for the institution's existing rails.

SEE IT LIVE

Read the interactive edition and grade the evidence yourself at runloci.com/fraud-ai-discovery-bottleneck. The companion paper, The Fraud Language Model, is at runloci.com/resource.