

Can Reputation Prevent Model Collapse? A Preregistered Test of Reputation-Weighted Data Filtering

Wulf A. Kaal, Ph.D.*

** Professor of Law, University of St. Thomas School of Law. ORCID 0009-0008-7840-1847. Correspondence: wulf@wulfkaal.com. Working paper, October 2026. The Gaussian outcomes of this study remain sealed at the time of posting. Model engines served as blind auditors of the design, the code, and this text, as Sections IV and VI describe. They are not independent peer review.*

Abstract

Intelligence models increasingly learn from data that earlier Intelligence models produced. When that loop repeats over generations, the rare cases disappear first: the unusual facts, minority patterns, and edge cases that make data valuable. This is known as model collapse. One proposed safeguard is to let a pool of validators decide which generated data is good enough to train on, and to give more weight to validators with a proven track record. This paper tests whether such a track record, earned once against a known answer key, actually helps a filter keep rare cases alive.

The test was designed, preregistered, and frozen before any result was read. On a simulated mixture of common and rare data clusters, the author compares a reputation-weighted validator pool with three benchmarks: no filter, keeping ten percent real data in every generation, and a standard likelihood filter. A second comparison checks whether any gain comes from the reputation weights themselves or merely from pooling several validators. The results exist but stay sealed until a replication of the original model-collapse experiment succeeds.

The question matters because the supply of fresh human-made data is finite, and the readings were fixed in advance, so the result informs either way. If reputation-weighted filtering helps here, it identifies a mechanism worth testing on real language models. If it fails, the paper shows where the idea breaks, with the same prominence. The paper follows on from the author's Reinforcement Learning from Reputation Feedback (RLRF) paper. It tests one use of that mechanism, as a filter on training data, not the mechanism as a whole.

Keywords: model collapse, synthetic data, recursive training, data filtering, rare-mode retention, reputation systems, validator pools, Reinforcement Learning from Reputation Feedback, RLRF, preregistration, blind analysis, mechanism design, Intelligence alignment, Computative Economics

JEL codes: C12, C15, C45, C90, D83, O33

Table of Contents

I. From Mechanism to Measurement	2
II. What the Literature Already Establishes	4
III. The Mechanism Under Test	6
IV. Design: Three Substrates, One Frozen Packet	8
V. Estimands and Inference	9
VI. Integrity Architecture	11
VII. Results	12
VIII. What Each Outcome Would License	12
IX. Limits	13
X. Conclusion	14
References	15

I. From Mechanism to Measurement

Generative models now train on the record their predecessors wrote. When a model is fit to samples drawn from an earlier model, and that loop repeats, the tails of the original distribution disappear first and later generations converge toward a point estimate with very small variance (Shumailov et al. 2024). The low-probability regions of the data are the first casualties. Image generators that consume their own output without enough fresh real data progressively lose quality or diversity (Alemohammad et al. 2024), and the scaling laws that made large models predictable change form once synthetic data enters the corpus (Dohmatob et al. 2024). As AI-generated content proliferates, it dilutes the diversity of the text available for later training, potentially degrading performance across generations (Kaal 2025).

The literature offers two established remedies. The first keeps real data in the loop. The second selects synthetic data that passes a verifier and can sustain performance where unfiltered recursion fails (Feng et al. 2025). Neither asks how to weight judgments behind a validity filter when several assessors with uneven, domain-specific reliability supply them.

The paper treats that question as one of mechanism design. Its working hypothesis is that collapse on the replacement protocol is, at least in part, a selection failure rather than a generation failure. Whoever writes the filter writes what the next generation learns. Reinforcement Learning from Reputation Feedback (RLRF) proposes an alignment mechanism in which an economically incentivized validation pool, whose members stake non-transferable reputation, replaces the human annotator panel (Kaal 2026b). Reputation there rises when a vote matches the pool's reputation-weighted consensus and falls more steeply when it does not (Kaal 2026b), and it is scoped by

domain, so that standing earned in one field confers no weight in another (Kaal 2026a). The RLRF paper does not test its mechanism against model collapse, and it poses no filter inside a recursive replacement loop as a remedy. That use of the mechanism is this paper's question.

The question is open on the evidence. The RLRF paper reports a cohort in which a party's record on checked work predicts its accuracy on unchecked work, but its best-record selector's margin over shipping whatever passes the checks in hand is 0.037, with an interval containing zero, and its own statement of limits calls RLRF an untested mechanism proposal (Kaal 2026b). Its separate selection experiment did not show that consensus-weighted selection outperformed equal-weighted selection, and its attribution test placed the observed difference at the 53.4th percentile of random reassignments of the weights (Kaal 2026b). A weighting rule earns no credit as a collapse remedy until it beats the simplest real-data remedy on a collapse protocol.

Three commitments discipline the answer. First, the paper names the limit of its own mechanism before any data exist. A validity filter judges whether an output is correct, not whether it is rare. Where rare content is as checkable as common content, such a filter can change rare composition only through the generator's own error profile. If collapse proceeds by under-sampling rare modes, no validity filter restores them, and a reputation-weighted filter is no exception. The experiment therefore includes an oracle validity filter as a premise. Second, the outcome is chosen to register the loss collapse inflicts. Total rare-region mass behaves close to a martingale under resampling: individual rare components drift and die while the aggregate stays near its starting value. The primary outcome therefore caps each rare component's retained mass at its original level and averages over components. A component that dies counts as a loss. A component that grows cannot offset it. Third, the design is built so that its author arranged not to read the outcome early, and so that specified later adjustments would be detectable.

The paper makes three contributions, each narrower than the mechanism it descends from. It poses, and preregisters a test of, whether domain-scoped reliability weights earned once on a disjoint key improve a validity filter under recursive replacement, against full replacement, real-data retention, and a likelihood filter, and against equal-weight pooling and a single verifier. It contributes a design for testing validity filters under recursion: a capped per-component outcome, an oracle premise that gates attribution, a bounded-exposure real-data comparator, and a coin-flip validator family whose leak or acceptance failure removes confirmatory status. And it implements the preregistration and blind-analysis standards strictly in its blinding and analysis controls, with two departures from its frozen procedure that Section IV discloses. That record is evidence about the conduct of the study, not about the filter.

Relation to the RLRF paper. This paper is a follow-on to the RLRF paper (Kaal 2026b). That paper proposes the mechanism, reports a measured cohort, a parameter sweep, and a separate selection experiment, and trains no policy. Relative to the parent paper, the contributions above come to three additions. The first is a preregistered and

frozen test of one use of the mechanism, as a filter on the training data of a recursive replacement loop. The second is an implementation of that test, which has been run on its Gaussian substrate with the outcomes sealed and unreported until the replication gate opens. The third is a statement, in Section II, of how the design bears on three of the parent paper’s open questions and what it leaves open. It does not test the parent mechanism as a whole, and it does not supply the end-to-end policy comparison the parent paper calls for.

II. What the Literature Already Establishes

Every component of the experiment has a literature older than the experiment. This section proceeds by concession: it names the closest work, states what that work establishes, and isolates what is left.

Which collapse. Shumailov et al. (2024) showed tails disappearing first across language models, variational autoencoders, and Gaussian mixtures. Theory located the boundaries: iterative retraining is stable when the initial model approximates the data distribution well enough, and the share of clean real data in each round is large enough (Bertrand et al. 2024), purely synthetic training cannot avoid collapse, while mixing in real data bounds the synthetic share that can (Seddik et al. 2024). And synthetic data cuts or narrows the tail of the data distribution and changes the form of scaling (Dohmatob et al. 2024). The regime matters most. Replacing real data with each generation’s synthetic data collapses, while accumulating real and synthetic data keeps error bounded (Gerstgrasser et al. 2024; Kazdan et al. 2025), and some argue that the catastrophic reading misdescribes realistic conditions (Schaeffer et al. 2026). This paper runs fixed-size replacement, the protocol on which collapse was first shown, and has no accumulation arm, so no result here bears on whether a filter adds anything once real data accumulates. Its outcome defines collapse as finite-sample extinction of low-weight modes.

Real data as a remedy. Retaining ten percent of the original data slows the damage (Shumailov et al. 2024), and accumulating all real data keeps test error bounded in the settings studied (Gerstgrasser et al. 2024; Kazdan et al. 2025), although even a synthetic share as small as 1 per 1,000 can stop larger training sets from helping in regression (Dohmatob et al. 2025). A filter that claims to address collapse must therefore beat real-data retention, not only unfiltered replacement.

Selection as a remedy. The closest prior work is Feng et al. (2025), who show that verifier-selected synthetic data can prevent collapse, with Gaussian-mixture theory and experiments comparing oracle and weak verifiers with self-selection and random selection. The paper concedes the core of that result: verifier selection is an established remedy, and the oracle ceiling and the Gaussian setting are established. Later work maps the limits. An imperfect external verifier drives the model toward the verifier’s own knowledge center (Yi et al. 2026). Synthetic data helps only when the loop is information-open, shaped by external signals that inject task information, and a coarse signal such as binary correctness is domain-general (Li, Sun, and Deng 2026).

Self-verification by internal confidence can prevent collapse even in fully synthetic regimes (Fu et al. 2025). Curation acts as implicit preference optimization that amplifies the curator’s biases (Ferbach et al. 2024), several heterogeneous rewards can instead preserve diversity (Falahati et al. 2026), and verifiers that see only fragmented slices of the reference can accelerate tail loss (Qiao et al. 2026). The pool tested here applies one domain-general validity bit, so what domain scoping can govern is whose judgment of that bit counts, not what the bit says.

Selection on rarity. Filters that select on surprise or entropy, not validity, work where they work because they see what a validity filter is forbidden to see: a high-surprise filter mitigates collapse about as well as human-data baselines (Gambetta et al. 2026), and an entropy-rate filter reports diversity gains where a log-probability filter gives no significant benefit (Mitchell 2026). Against that literature, a null here is a literature rather than a caveat.

What the weight already is. Weighting assessors by estimated reliability is old (Dawid and Skene 1979; Whitehill et al. 2009), and multiplicative weights supply its online form (Littlestone and Warmuth 1994; Freund and Schapire 1997). Peer prediction makes honest reporting an equilibrium without a key (Miller, Resnick, and Zeckhauser 2005; Prelec 2004). The RLRF paper concedes its weight is a member of this family (Kaal 2026b). In this experiment, the concession is stronger: reputation is settled once against a key before training, so the pool is supervised reweighting from a disjoint key. The paper does not compare it with a key-free latent-variable aggregator.

Goodhart. Optimizing a proxy eventually degrades the target (Gao, Schulman, and Hilton 2023), and curated loops optimize their curator (Ferbach et al. 2024). This is the channel the RLRF program claims reputation can resist. This study does not open it: its validators are stationary, its stake has no external value, and its weights are fixed before generation 1.

Preregistration. Preregistration, registered reports, and blind analysis are established standards (Nosek et al. 2018; Chambers and Tzavella 2022; MacCoun and Perlmutter 2015). The paper claims no invention there.

What the parent paper already reports. Four passages of the RLRF paper bear on the question here, and the present study is distinct from each.

First, the RLRF paper proposes, without implementing or testing it, a training format in which candidates that received a UP resolution with high reputation-weighted consensus become training targets, and it lists the resulting flywheel, in which better pools yield better training data, as untested at every stage (Kaal 2026b). That format is the parent paper’s nearest construct to a data filter. It weights by pool consensus rather than by a key, sits outside any collapse protocol, and has not been run. The filter here weights by a key, operates inside a fixed replacement protocol, is scored by a capped rare-component outcome, and has been run on its Gaussian substrate, with outcomes sealed.

Second, the RLRF paper's Experiment T compared consensus-weighted with equal-weighted selection by a panel of seven small open instruct models choosing among candidate responses on a two-constraint instruction-following task that the paper calls easy (Kaal 2026b). The observed difference was about 0.002, its sign reversed in the paper's own sensitivity check, its interval contained zero, attribution was negative, and the paper states that the experiment does not test the mechanism. Both experiments ask whether weighting assessors adds anything to pooling them, but they differ on every axis that defines a test. Experiment T weights by consensus, selects once, and uses model assessors whose reliability is not set by design. Here the weights come from a key on disjoint items, the filter governs the training data of fifty generations of recursive replacement, and the validators are simulated with a reliability structure set by design, so that in two of the three required families the weights could matter, as a feature of the design and not an expectation about the outcome. Experiment T is not revisited, and a result in one experiment would not answer the question posed by the other.

Third, the RLRF paper leaves open whether reputation weights come to track validator accuracy rather than agreement with the pool, and whether weights earned on keyed tasks transfer to unkeyed ones (Kaal 2026b). The ladder of Section V asks a narrower form of both. Its third rung asks whether the weights track accuracy on held-out keyed items, but every update here is driven by the key, so the contrast with agreement with the pool never arises. Its fourth rung asks whether weights earned on keyed rows improve the validity of candidates that no key settles, but the simulated validators' accuracy depends only on the tag, so the keyed rows are informative about the candidates by construction, and the rung checks whether the pipeline carries that information into what the pool accepts, not whether real validators would show the same transfer. In the homogeneous family, where the weights have nothing to find, the rung is a bound on attribution and not a prediction. The parent paper's measured cohort concerns the records of real model parties rather than pool weights, and its parameter sweep uses simulated validators that the paper itself says cannot validate the mechanism (Kaal 2026b). The simulated families here carry the same limit, and their errors are independent by construction, so validator independence is not examined either.

Fourth, the RLRF paper names, as prior to the six open problems it then lists, an end-to-end comparison in which policies are trained under a reputation-weighted signal, under an annotator or model-rater baseline, and under a visible-check selector, are evaluated on held-out work, and have the reputation weighting ablated separately (Kaal 2026b). The present design is not that comparison. Its pool supplies no reward signal and trains no policy, and its equal-weight contrast ablates the weights inside a data filter rather than inside policy training, so it does not supply the ablation that paper asks for. It also runs no Full Vote, the staked and slashable second round of the RLRF protocol, and its single keyed update is not a stake settlement.

What is left is a comparative identification question none of this work answers: on the replacement protocol, under an outcome that counts the death of a rare component,

with every arm denied rare-set membership, does a validity filter whose validators carry domain-scoped weights earned on a disjoint key retain more of the rare components than full replacement, real-data retention, and a likelihood filter, and does any advantage survive ablation to equal-weight pooling and to a single verifier?

III. The Mechanism Under Test

The two pools at the center of the design, the reputation-weighted pool and the equal-weight pool, apply every rule in common except one: the weight a validator’s vote carries on a domain, and whether that weight was earned. Every rule below is the frozen design’s rule for substrate G, the only substrate on which the filter is tested.

What any filter may know. Validators judge validity only. No validator, filter, or score is asked about rarity, novelty, frequency, diversity, or coverage. The outcome is computed by an evaluator alone, from a sealed reference holding rare-set membership and component identity, which no filter reads. The oracle alone receives one bit per candidate from it, the candidate’s true validity verdict. A point x is valid when its squared Mahalanobis distance to the nearest component mean satisfies $d_M^2(x, \mu_k) \leq \chi_{2,0.95}^2 \approx 5.991$, so validity is balanced between rare and common regions to within 0.02, as a check before any run confirmed. Validators also see a tag, the nearest of four hash-drawn centers, chosen so that $|P(\text{rare}|t) - P(\text{rare})| \leq 0.04$ for every tag t .

The pool. Five validators are seated per candidate from a roster of twenty. Each votes independently. A candidate c is accepted when its weighted valid share $s(c) = \sum_i w_i v_i(c)$ exceeds $\theta = 0.5$, where $v_i(c) \in \{0,1\}$ is validator i ’s vote. A candidate with $|s(c) - \theta| \leq 0.1$ is contested and receives four more validators, and the recomputed share binds. Each vote stakes one reputation unit, a constant with no external value. The RLRF paper specifies a two-round protocol with argumentation between rounds (Kaal 2026b). This study keeps the weighted vote and the stake and replaces the two rounds with a single assessment and a threshold-triggered second look. Every drawing arm samples its own 4,000 candidates per generation and trains on 2,000 rows. Validator compute is not equal across arms: the pool seats five validators per candidate, nine when contested, and the single verifier one. A threshold arm that accepts too few fills the shortfall uniformly from rejected candidates, and an arm whose minimum over generations of the seed-mean fill-free share, $\min(\text{accepted}, N)/N$ with $N = 2,000$ training rows per generation, falls below $a_{\min} = 0.98$ in a family has its tests there invalidated.

The weight. The update is equation (3) of the RLRF paper, an asymmetric multiplicative rule. On tag t , reputation updates as $r_{i,t} \leftarrow r_{i,t}(1 + \alpha)$ when the vote matches the revealed key and $r_{i,t} \leftarrow r_{i,t}(1 - \gamma)$ when it does not. The RLRF paper requires only

$\gamma > \alpha$, and this design fixes $\alpha = 0.05$ and $\gamma = 0.10$. Each reputation starts at $r_{i,t}^{(0)} = 1$,

and after each update the floor $r_{i,t} \geq 0.01$ is applied. Weights are $w_i = r_{i,t} / \sum_j r_{j,t}$ over

the seated validators. Any weight above 0.5 is capped at 0.5, and the excess is redistributed among the other seated validators in proportion to their weights, so that

$\sum_i w_i = 1$. The RLRF paper states the rule for unkeyed events and allows keyed

settlement. This study uses only keyed settlement, once, after the generation-0 fit on real data and before generation 1. No candidate that enters training ever updates a reputation. The keyed record holds 192 real rows, stratified so that it teaches the pool who judges validity well on which tag and nothing about where the rare components lie beyond the bounded tag leakage. Those $K_G = 192$ rows are the pool's entire exposure to real data, below the $[0.1N] = 200$ real rows the ten percent arm receives in one generation and far below the $N(1 - 0.9^{50}) \approx 1,990$ distinct real rows it trains on over the run.

The validators. Twenty simulated validators vote the true validity bit, flipped independently with probability one minus their accuracy on the candidate's tag. Family U is homogeneous at 0.75. In family H each validator is reliable at 0.90 on two hash-drawn tags and at 0.60 on the other two, so domain-scoped reputation has something to find. Family Mn has five validators at 0.90 and fifteen at 0.70, a competent minority. The three share a mean accuracy of 0.75. Family Z votes by coin flip. It is a negative control: a coin-flip pool knows nothing about validity, and if it meets the registered leak criterion or fails its acceptance rule, both substrates lose confirmatory status.

The comparators. Three are confirmatory. Full replacement is the collapse condition and also the equal-budget random-selection control. The ten percent arm adapts the published retention control to this substrate, training each generation on 200 real rows freshly drawn from the 2,000 generation-0 real rows plus 1,800 candidates, fit by the same procedure as every other arm. A likelihood filter keeps the 2,000 most likely candidates under the current generator. Three are secondary. The equal-weight pool is the only contrast that can attribute a difference to the weights. A single verifier per tag measures pooling and weighting together. A bounded-exposure arm retains 192 fixed real rows every generation and is tested for non-inferiority only, because a keyed row and a real row need not carry the same information. Two reference arms gate interpretation: an oracle that accepts exactly the valid candidates, and a no-recursion arm trained on fresh real data, which defines whether collapse occurred.

The pool under test omits most of what makes RLRF an institution. It has no economy, no strategy, no adversary, no deliberation, and no reputation dynamics after the single settlement.

IV. Design: Three Substrates, One Frozen Packet

The design binds in one file, frozen by hash only after twenty-two rounds of blind audit by two model engines, the last of which both passed with no findings. Every harness and analysis step refuses to run against a design whose hash differs. Every random draw derives from a named hash template, so no draw depends on anything outside its listed fields.

Substrate P, the replication gate. It replicates the published language-model protocol: the 125-million-parameter causal language model the published protocol specifies (Shumailov et al. 2024), fine-tuned on a standard Wikipedia text corpus (Merity et al. 2016), full replacement for five epochs against ten percent retention for ten, generations 0 through 9, five runs each. The filter never enters it. The frozen pass rule requires, on the five-run means from generation 0 to 9, a full-replacement perplexity rise $\Delta_{FR} \geq 4.0$ with $\Delta_{FR} - \Delta_{R10} \geq 2.0$, where Δ_{R10} is the same rise under ten percent retention. The thresholds sit at fractions of the reported 8-point change so that the rule does not depend on which of two published baselines, 20 or 34, was meant. A miss blocks reading anything else.

Substrate G, the Gaussian mixture. Ten components in two dimensions, placed by hash at least 6 units apart on the square from -12 to 12. Four are assigned to the rare set by hash after placement, at weight 0.015 each. All four pre-run checks passed: validity balance, a rare-region mass of 0.060, rare cells of about 0.015 each, and tag leakage on the 160th center set. The study runs 200 seeds with a reserve of 40, fifty generations each. All 200 seeds completed in every family, with no fit failure and no reserve seed used. The outcomes sit in a sealed store.

Substrate F, declared procedure-invalid. A synthetic fact corpus was to be the fact-corpus test of the filter. The operator declared it procedure-invalid before any run or reading. No pre-run rule fired, and no written reason was recorded. Its nine registered tests stand at p equal to 1 with no other label, and the paper draws no inference from it. The consequence is plain: the filter is tested on one substrate, a two-dimensional mixture with simulated validators.

Two departures. The preregistered pilot, three seeds and two generations meant to confirm that every arm completes, was never run, and the confirmatory seeds ran without it. Their later completion does not cure the skipped gate. The substrate F declaration is the second departure. Both qualify every reading in Section VIII.

V. Estimands and Inference

The estimand. For the set R of four rare components, with $p_{50}(k)$ the probability the generation-50 model assigns to component k 's cell and m_k that cell's frozen mass, the outcome is

$$R_{50} = \frac{1}{|R|} \sum_{k \in R} \min\left(1, \frac{p_{50}^{(k)}}{m_k}\right).$$

Only generation 50 is confirmatory. Each final-generation estimate uses $L_G = 4 \times 10^6$ draws under common random numbers. With $m_{cell,min} = 0.01$ the frozen floor that every rare-cell mass was required to clear before any run, that bounds each component's standard error by $0.5/(\sqrt{L_G} m_{cell,min}) = 0.5/(2,000 \times 0.01) = 0.025$, a quarter of the smallest effect of interest. Uncapped rare mass is reported beside it, supports no claim, and shows what the cap changed.

Uncertainty and labels. The unit is the seed, and every contrast is a paired seed-level difference within a validator family. A seed bootstrap with $B = 10,000$ replicates gives percentile intervals and the two-sided p-value $p = \frac{1}{B} \#\{b: |d_b - d| \geq |d|\}$ for an observed contrast d , where d_b is the contrast in bootstrap replicate b (Efron and Tibshirani 1993). A contrast is superior when its Holm-adjusted p-value is below 0.05 (Holm 1979), its 95 percent interval lies above zero, and its point estimate satisfies $\hat{d} \geq \delta = 0.10$. It is negative when the adjusted p-value is below 0.05 and the interval lies below zero, a substantive null when its 90 percent interval lies within $[-\delta, \delta]$, and inconclusive otherwise. The effect sizes, $\delta = 0.10$, a planning alternative $\delta_{alt} = 0.15$, and a non-inferiority margin $\delta_{NI} = 0.05$, were fixed by the operator before any pilot.

Families. The confirmatory family of twelve holds the pool against full replacement, the ten percent arm, and the likelihood filter in each of the three required families, plus the three substrate F tests at p equal to 1. Three secondary families hold the reputation-over-pooling, single-verifier, and non-inferiority contrasts. A failure-mode family of twelve holds the likelihood filter, the equal-weight pool, and the reputation pool, each against full replacement. Family Z sits outside every Holm family.

Gates. Before any label, any of the three confirmatory contrasts computed in family Z with a 95 percent interval above zero and $\hat{d} \geq 0.10$ is a leak, and a family Z minimum seed-mean fill-free share below 0.98 for the single verifier, the equal-weight pool, or the reputation pool is an acceptance failure. Either removes confirmatory status on both substrates. After labels, four conditions restrict claims. The collapse rung requires the no-recursion arm to beat full replacement by at least 0.10 with an interval above zero, and a planning calculation put that gap near 0.4, an expectation stated before any run, not a result. The oracle premise requires the oracle to clear the same bar in each family, and if it fails, no positive label is attributed to validity filtering. The coverage gate requires the pool's seed-mean coverage, the fraction of rare components retaining at least half their original mass, to be at least one half, with the 95 percent lower bound of its coverage minus full replacement's above zero. The key cap holds by construction at 192 against 200.

The ladder. A null is more useful when its location is known. Six rungs, read in order, report the first link not demonstrated among the five that carry a pass rule: collapse, the oracle premise, whether the weights tracked reliability on held-out keyed items, measured by a Spearman correlation (Spearman 1904) whose 95 percent lower bound must exceed 0.3, whether the weights improved the validity of what the pool accepted, a descriptive composition rung that never stops the ladder, and retention itself. A failed rung means a link was not demonstrated, not that the effect is absent.

Power. A positive reading on substrate G needs nine superior labels. For a contrast with paired seed-level standard deviation s_d over S seeds, $se = s_d/\sqrt{S}$. Under a normal approximation and a true effect δ_{alt} , the label power is

$$\Phi\left(\frac{\delta_{alt} - \max(\delta, z_{1-0.05/24} se)}{se}\right), \quad z_{1-0.05/24} \approx 2.865,$$

where $z_{1-0.05/24}$ is the critical value at the most demanding Holm step and Φ is the standard normal distribution function. The design takes the product of the nine single-test powers as the conjunction power. At $S = 200$ and $\delta_{alt} = 0.15$, the break-even s_d for a conjunction power of 0.80 is about 0.36, a figure that holds only when the comparator's retention is at most 0.85. If the ten percent arm's retention exceeds 0.90 in a family, the pool cannot be labeled superior to it there, and no positive reading is possible. No standard deviation was known before the runs, so the design fixes a masked release: after the gate opens, the analysis releases only each test's variance of paired differences, withholds the mean, and reports the realized power first. If the realized conjunction power is below 0.80, the substrate is reported as underpowered, with the same prominence as any label. That power covers the labels only, not the gates.

VI. Integrity Architecture

A preregistration binds an author only to the extent that departures from it would be visible. The audits came first. The design took twenty-two rounds. The substrate G harness and analysis were audited together over five rounds, and the harness passed in the fifth with minor findings, which were queued for a later commit, to be audited before unsealing. The harness file has not changed since, and the confirmatory seeds ran under a later commit that differs from it only in documentation and the pre-run record. The analysis was then audited alone and passed with no findings in its thirteenth round, before any outcome was read. The substrate P harness passed in its seventh. Each of the ten sections of the longer working draft from which this text is condensed was audited by model engines before it was committed, with the exceptions Section IX notes. This condensed text is audited in its own rounds, and the record of those rounds is kept apart from those of the ten sections. The engines are not independent peer review: they may share failure modes with each other and with the author's tools.

Outcomes are written only to a sealed store, readable only by their owner, with a manifest of digests for every file. The seal is a commitment made visible, not a lock: the operator can open the files. The analysis checks each manifest's provenance, checks every sealed file against its digest before reading it, and records a digest of every file the variance release reads, refusing to unseal if any changed. It refuses unless the design hash is accepted, the tree is clean and matches its commit byte for byte, and the code differs from the audited commits only in a named allowlist. It scans for any module that could shadow its libraries before loading one, and it reads the audited commits it trusts from a local file outside version control, where no commit can move them. The store stays sealed until a substrate P record meets the frozen pass rule.

The design also fixes, before any data, the sentences the paper may write for each outcome and the conditions for each, and it bars claims about staking incentives, slashing, collusion, Sybil resistance, deployed systems, natural web text, frontier-scale models, or a solution to collapse. The architecture shows that the author arranged not to read the outcome early and that specified later adjustments would be detectable. It does not show that the filter works, and it does not prove the outcome was never read. The Gaussian outcomes also already exist, written before any venue reviewed the design, so the paper does not claim untouched Stage 1 status.

VII. Results

No result is reported yet. Substrate P is running: as of 11 October 2026, one of its ten arm-runs has completed and no record exists. Substrate G has finished, and its outcomes sit sealed. Substrate F is procedure-invalid. The results section of the working draft is written in advance as a reporting shell: it fixes the order of reporting, the P verdict first, then procedure validity, then the nine confirmatory labels, the secondary and failure-mode labels, the gates and ladder, and the secondary outcomes and costs, with every number left empty until the frozen analysis fills it. A reader will be able to compare the finished section against the version written before any outcome was read.

VIII. What Each Outcome Would License

Each reading below stays inside a sentence the design registered, with its conditions. A null, a negative, and an inconclusive label are each reported with the same prominence as a positive.

If the replication fails, the paper reports that the published result did not reproduce under the frozen rule, substrate G stays sealed, and no filter result is reported. The study could not say whether the implementation, its fixed choices, or the published result accounts for the miss.

If confirmatory status is lost through a family Z leak or acceptance failure, every label is reported without confirmatory weight, and no confirmatory reading is issued.

A positive reading is issued only if all nine confirmatory tests are superior, the collapse rung, oracle premise, coverage gate, and the acceptance gate of Section III, which

binds every threshold arm including the single verifier and the equal-weight pool, hold in every family, no family Z leak or acceptance failure occurred, and the key cap holds. It would say that, on substrate G, recursive training on synthetic data accepted by a validity pool had higher capped per-component rare-mode retention, over rare components, after 50 generations than full replacement, ten percent real retention, and a likelihood filter, at the same candidate budget, under a measure in which a gain in one component cannot offset a loss in another. Like every reading that names the ten percent arm, it would state the pool's key exposure of 192 real rows against the roughly 1,990 distinct real rows that arm trained on. Equal candidate budgets do not mean equal validator compute, which the cost table reports. The positive reading would not say the weights mattered. That needs the attribution sentence, issued only if the pool's contrast against full replacement and the reputation pool's contrast against the equal-weight pool are both superior in every family, the homogeneous one included, the oracle premise and ladder rungs 3 and 4 pass in every family, and the key cap holds. Otherwise, if the reputation contrast is null or negative in any family, the paper says only that any difference is not attributable to reputation over equal weighting, and otherwise, if it is inconclusive in any family, only that attribution is unresolved.

A null on a contrast in which the pool is the first arm would mean only that, on this protocol as specified, the pool was not shown to retain more than that comparator. The design does not choose among the reasons: the validity-rarity limit, low power, validator noise, an ineffective weighting, the fill rule, or the substrate. A substantive null says the difference is at most 0.10 in magnitude.

A negative on such a contrast would mean the pool retained less than that comparator. If the likelihood filter, the equal-weight pool, and the reputation pool each carry the negative label against full replacement in every family, the paper issues the registered failure-mode sentence: each had lower capped retention at generation 50 than full replacement. The design named that pattern in advance. It is a registered negative result, not an implementation failure.

If the oracle premise fails, the reading depends on how. If the oracle contrast misses the registered margin, perfect validity information was not shown to help at that margin, which is consistent with the validity-rarity limit without proving it. If instead the oracle fails its acceptance gate, part of its training set was filled from rejected candidates, and that event says nothing about whether perfect validity information helps. Either way, no positive label can be attributed to validity filtering, weighted or not.

Every reading above carries the two departures: the skipped pilot and the operator's declaration on substrate F.

IX. Limits

The filter is tested on one substrate, a two-dimensional mixture, and never inside a language-model loop. Its validators are simulated, and the families are, in the design's words, the result. They err independently given their accuracy, which depends only on

the tag, so shared lineage, correlated errors, and error that depends on whether content is rare are excluded by construction, and the study supplies no evidence about correlated assessors or about reputation concentrating in a lineage. Validity is balanced between rare and common content by construction, so the study says nothing about settings where rare content is harder to check. The weight is settled once and compared only with equal weighting and a single verifier, never with a key-free aggregator (Dawid and Skene 1979). Stake has no external value, and the study cannot speak to incentives, collusion, Sybil resistance, or Goodhart pressure. Nine superior labels set a high bar, so a real but modest effect may be reported as not positive. The two departures, the skipped pilot and the operator's declaration on substrate F, qualify every reading. And in the final audit rounds of the last three sections of the working draft, one audit family's command-line engines returned no verdict because their usage balance was exhausted, so those rounds hold two passing verdicts rather than three, and they left identified low-severity findings unadopted, as their audit records state.

The design assigns stake size, slashing cost, and collusion to a second study. The paper would add tests of strategic adaptation and Goodhart pressure, a test inside a language-model loop on natural text, a comparison with a key-free aggregator and with filters that select on surprise or entropy, and later work on scale. None is promised. Each would need its own design, preregistration, and audit.

One follow-on study would bear most directly on the parent paper, by combining three changes, each aimed at a limit stated above. It would replace the simulated validators with model validators drawn from several lineages and settled on keyed items, so that correlated errors and the concentration of reputation in a lineage could be measured rather than excluded by construction. It would add, beside the keyed pool, an arm that reproduces the relevant elements of the parent paper's proposed answer-generation format. In that arm, reputation would update against realized pool resolutions on unkeyed events, and candidates that received an UP resolution with high reputation-weighted consensus would become training targets. Held-out keyed items would then measure whether the resulting weights track accuracy or only agreement with the pool, which is the parent paper's second open problem, under recursion. And it would run the filter inside a language-model loop on natural text. Such a study would still omit economic stake and deliberation, and so would test the parent mechanism only closer to its published form, not as a whole. Nor would it be the end-to-end policy comparison the parent paper names as prior to all six of its numbered open problems, which would remain a separate study. This follow-on is the paper's suggestion and is separate from the second study that the design names, which would take up stake size, slashing cost, and collusion. Like the items above, it is not promised.

X. Conclusion

The question was never whether reputation can be described as a fitness function for recursive training. It is whether a domain-scoped reputation weight, settled once against a key, measurably improves a validity filter under recursive replacement, on the substrate where the test can be run. The question matters because the supply of fresh

human-made data is finite, and a filter that keeps rare content alive across generations would let synthetic data stand in for part of it.

This paper reports no results yet. The simulated runs are complete, but their outcomes stay sealed until the replication of the published collapse result succeeds. What the paper contributes now is a way of asking: a ceiling named before any data, an outcome that counts the death of a rare component, a separation of what a pool does from what its weights do, sentences fixed before the outcome, and a record of where the study departed from its own procedure.

The answer may not come. If the replication fails, confirmatory status is lost, or substrate G proves procedure-invalid, the narrow hypothesis remains unexamined, and the paper does not establish the working hypothesis. If the seal opens, the results will be added in a revised version. A positive reading would identify a mechanism worth testing on real language models. A null or negative reading would be reported with the same prominence, and the ladder would show which link failed. Either reading will answer only the narrow question.

The paper follows on from the RLRF paper (Kaal 2026b) and tests one use of that mechanism, as a filter on training data, not the mechanism as a whole. A mechanism that has not been tested has not been shown to work. This paper tests one version of it, under a design that fixed its readings in advance and makes specified later adjustments detectable, and in which the author arranged not to read the outcome early.

References

- Alemohammad, Sina, Josue Casco-Rodriguez, Lorenzo Luzi, Ahmed Imtiaz Humayun, Hossein Babaei, Daniel LeJeune, Ali Siahkoohi, and Richard G. Baraniuk. 2024. "Self-Consuming Generative Models Go MAD." In *The Twelfth International Conference on Learning Representations* (ICLR 2024). arXiv:2307.01850.
- Bertrand, Quentin, Avishek Joey Bose, Alexandre Duplessis, Marco Jiralerspong, and Gauthier Gidel. 2024. "On the Stability of Iterative Retraining of Generative Models on Their Own Data." In *The Twelfth International Conference on Learning Representations* (ICLR 2024). arXiv:2310.00429.
- Chambers, Christopher D., and Loukia Tzavella. 2022. "The Past, Present and Future of Registered Reports." *Nature Human Behaviour* 6 (1): 29-42. <https://doi.org/10.1038/s41562-021-01193-7>.
- Dawid, A. P., and A. M. Skene. 1979. "Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm." *Journal of the Royal Statistical Society, Series C (Applied Statistics)* 28 (1): 20-28. <https://doi.org/10.2307/2346806>.

- Dohmatob, Elvis, Yunzhen Feng, Arjun Subramonian, and Julia Kempe. 2025. "Strong Model Collapse." In *The Thirteenth International Conference on Learning Representations* (ICLR 2025). arXiv:2410.04840.
- Dohmatob, Elvis, Yunzhen Feng, Pu Yang, François Charton, and Julia Kempe. 2024. "A Tale of Tails: Model Collapse as a Change of Scaling Laws." In *Proceedings of the 41st International Conference on Machine Learning* (ICML 2024), PMLR 235: 11165-11197. arXiv:2402.07043.
- Efron, Bradley, and Robert J. Tibshirani. 1993. *An Introduction to the Bootstrap*. New York: Chapman and Hall. <https://doi.org/10.1201/9780429246593>.
- Falahati, Ali, Mohammad Mohammadi Amiri, Kate Larson, and Lukasz Golab. 2026. "Curated Synthetic Data Doesn't Have to Collapse: A Theoretical Study of Generative Retraining with Pluralistic Preferences." In *Proceedings of the 43rd International Conference on Machine Learning*, PMLR 306: 28690-28725. <https://proceedings.mlr.press/v306/falahati26a.html>.
- Feng, Yunzhen, Elvis Dohmatob, Pu Yang, François Charton, and Julia Kempe. 2025. "Beyond Model Collapse: Scaling Up with Synthesized Data Requires Verification." In *The Thirteenth International Conference on Learning Representations* (ICLR 2025). arXiv:2406.07515.
- Ferbach, Damien, Quentin Bertrand, Avishek Joey Bose, and Gauthier Gidel. 2024. "Self-Consuming Generative Models with Curated Data Provably Optimize Human Preferences." In *Advances in Neural Information Processing Systems 37* (NeurIPS 2024): 102531-102567. <https://doi.org/10.52202/079017-3256>.
- Freund, Yoav, and Robert E. Schapire. 1997. "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting." *Journal of Computer and System Sciences* 55 (1): 119-139. <https://doi.org/10.1006/jcss.1997.1504>.
- Fu, Shi, Yingjie Wang, Yuzhu Chen, Li Shen, and Dacheng Tao. 2025. "Self-Verification Provably Prevents Model Collapse in Recursive Synthetic Training." In *Advances in Neural Information Processing Systems 38* (NeurIPS 2025): 36101-36154. <https://doi.org/10.52202/085713-1213>.
- Gambetta, Daniele, Gizem Gezici, Fosca Giannotti, Dino Pedreschi, Alistair Knott, and Luca Pappalardo. 2026. "Learning by Surprise: Adaptive Mitigation of Model Collapse in Large Language Models." *ACM Transactions on Intelligent Systems and Technology*. <https://doi.org/10.1145/3828663>.
- Gao, Leo, John Schulman, and Jacob Hilton. 2023. "Scaling Laws for Reward Model Overoptimization." In *Proceedings of the 40th International Conference on Machine Learning* (ICML 2023), PMLR 202: 10835-10866. arXiv:2210.10760.
- Gerstgrasser, Matthias, Rylan Schaeffer, Apratim Dey, Rafael Rafailov, Henry Sleight, John Hughes, Tomasz Korbak, et al. 2024. "Is Model Collapse Inevitable? Breaking the Curse

- of Recursion by Accumulating Real and Synthetic Data.” In *First Conference on Language Modeling* (COLM 2024). arXiv:2404.01413.
- Holm, Sture. 1979. “A Simple Sequentially Rejective Multiple Test Procedure.” *Scandinavian Journal of Statistics* 6: 65-70. <https://doi.org/10.2307/4615733>.
- Kaal, Wulf A. 2025. “Artificial Intelligence The Final Frontier.” SSRN 5095633. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5095633.
- Kaal, Wulf A. 2026a. “Evolution of Domain-Specific Reputation Systems: From Binary Validation to Citation-Weighted Knowledge Attribution.” SSRN 6192998. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=6192998.
- Kaal, Wulf A. 2026b. “Reinforcement Learning from Reputation Feedback: An Alignment Mechanism Grounded in Computative Economics.” SSRN 7456999. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=7456999.
- Kazdan, Joshua, Rylan Schaeffer, Apratim Dey, Matthias Gerstgrasser, Rafael Rafailov, David L. Donoho, and Sanmi Koyejo. 2025. “Collapse or Thrive: Perils and Promises of Synthetic Data in a Self-Generating World.” In *Proceedings of the 42nd International Conference on Machine Learning*, PMLR 267: 29469-29494. <https://proceedings.mlr.press/v267/kazdan25a.html>.
- Li, Hanyu, Zhengqi Sun, and Xiaotie Deng. 2026. “An Information-Theoretic Criterion for Efficient Data Synthesis.” In *Proceedings of the 43rd International Conference on Machine Learning*, PMLR 306: 70064-70075. <https://proceedings.mlr.press/v306/li26fp.html>.
- Littlestone, Nick, and Manfred K. Warmuth. 1994. “The Weighted Majority Algorithm.” *Information and Computation* 108 (2): 212-261. <https://doi.org/10.1006/inco.1994.1009>.
- MacCoun, Robert, and Saul Perlmutter. 2015. “Blind Analysis: Hide Results to Seek the Truth.” *Nature* 526 (7572): 187-189. <https://doi.org/10.1038/526187a>.
- Merity, Stephen, Caiming Xiong, James Bradbury, and Richard Socher. 2016. “Pointer Sentinel Mixture Models.” arXiv:1609.07843.
- Miller, Nolan, Paul Resnick, and Richard Zeckhauser. 2005. “Eliciting Informative Feedback: The Peer-Prediction Method.” *Management Science* 51 (9): 1359-1373. <https://doi.org/10.1287/mnsc.1050.0379>.
- Mitchell, Lewis. 2026. “No Model Required: Text Entropy Rate Filtering Mitigates Iterative Fine-Tuning Collapse.” arXiv:2610.01493.
- Nosek, Brian A., Charles R. Ebersole, Alexander C. DeHaven, and David T. Mellor. 2018. “The Preregistration Revolution.” *Proceedings of the National Academy of Sciences* 115 (11): 2600-2606. <https://doi.org/10.1073/pnas.1708274114>.

- Prelec, Dražen. 2004. "A Bayesian Truth Serum for Subjective Data." *Science* 306 (5695): 462-466. <https://doi.org/10.1126/science.1102081>.
- Qiao, Xinbao, Xianglong Du, Wei Liu, Jingqi Zhang, Peihua Mai, Meng Zhang, and Yan Pang. 2026. "When Sample Selection Bias Precipitates Model Collapse." In *Proceedings of the 43rd International Conference on Machine Learning*, PMLR 306: 101222-101268. <https://proceedings.mlr.press/v306/qiao26c.html>.
- Schaeffer, Rylan, Joshua Kazdan, Alvan Caleb Arulandu, and Sanmi Koyejo. 2026. "Position: Multiple Definitions & Unrealistic Assumptions of Model Collapse Distract from Real World Threats." In *Proceedings of the 43rd International Conference on Machine Learning*, PMLR 306: 171990-172003. <https://proceedings.mlr.press/v306/schaeffer26a.html>. Preprint: "Position: Model Collapse Does Not Mean What You Think," arXiv:2503.03150.
- Seddik, Mohamed El Amine, Swei-Wen Chen, Soufiane Hayou, Pierre Youssef, and Merouane Debbah. 2024. "How Bad Is Training on Synthetic Data? A Statistical Analysis of Language Model Collapse." In *First Conference on Language Modeling (COLM 2024)*. arXiv:2404.05090.
- Shumailov, Ilia, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. 2024. "AI Models Collapse When Trained on Recursively Generated Data." *Nature* 631 (8022): 755-759. <https://doi.org/10.1038/s41586-024-07566-y>.
- Spearman, C. 1904. "The Proof and Measurement of Association between Two Things." *The American Journal of Psychology* 15 (1): 72-101. <https://doi.org/10.2307/1412159>.
- Whitehill, Jacob, Paul Ruvolo, Tingfan Wu, Jacob Bergsma, and Javier Movellan. 2009. "Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise." In *Advances in Neural Information Processing Systems 22 (NIPS 2009)*.
- Yi, Bingji, Qiyuan Liu, Yuwei Cheng, and Haifeng Xu. 2026. "Escaping Model Collapse via Synthetic Data Verification: Near-term Improvements and Long-term Convergence." In *The Fourteenth International Conference on Learning Representations (ICLR 2026)*. arXiv:2510.16657.