# Who Wrote the Final Decision? Auditing Decision Provenance in Small Language Model Agents

**Gi Young Lee**[1*], **Byounggook Cho**[2*], **Jong Won Baek**[1], **Chansam Park**[1]
**Yohan Jung**[1], **Tae Hyung Kim**[1], **Kyoung Jae Won**[2†]

[1] BioNexus Inc. (bionexus.kr)
Gwanggyo Business Center, 156 Gwanggyo-ro, Yeongtong-gu,
Suwon-si, Gyeonggi-do, Republic of Korea

[2] Department of Computational Biomedicine,
Cedars-Sinai Medical Center (cshs.org)
750 N San Vicente Blvd, West Hollywood, CA, 90024 USA

## Abstract

Current benchmarks for small language models (SLMs) typically consider multiple factors, including task success, energy use, latency, and memory usage. Among these, task success is often measured from the final pipeline output, even when the outcome is influenced by multiple steps, such as validators, fallbacks, and ranking rules. This makes it difficult to determine whether the final outcome actually reflects the model's performance. To address this problem, we audit all 15 decision points in our drug-repurposing agent pipeline across four model sizes (27B, 9B, 4B, and 2B parameters). Our analysis reveals that end-to-end results can be misleading in assessing the performance of SLMs. While five of eight conditions yielded the same aggregated recommendation, run-to-run agreement drops from 0.82 to 0.52 and citation rejection increases drastically from 2.8% to 100.0%. Across the 2,200 pre-debate match verdicts, the model-path answer remained unchanged in 54%, while deterministic code—guards that check and may correct verdicts (K), and deciders that select winners by fixed rules (F)—was the last surviving writer in 46% (38–57% across model sizes). Aggregation obscures much of the remaining variation. We also find that the aggregator can favor candidates that reach the final stage rather than candidates that win more tournaments. Changing the aggregation policy removes the apparent differences between the 9B and 27B models but not the difference for 2B. We identify and fix nine evaluation defects, including four that introduce model- or serving-configuration-dependent effects. These findings demonstrate that rigorous SLM evaluation requires auditing intermediate decision nodes rather than relying exclusively on final pipeline outputs.

## 1 Introduction

Efficiency-aware evaluations of small language models (SLMs) pair task success with energy, latency, and memory. The costs are measured directly; task success is not. It is read off the final output of an agent pipeline and then attributed to the model, even though validators, retries, fallbacks, fixed rules, and ranking or voting steps all shape that output.

*These authors contributed equally.
†Corresponding author: KyoungJae.Won@cshs.org.

40th Conference on Neural Information Processing Systems (NeurIPS 2026). Workshop: SLMs for Agentic Systems, Paris, France, 2026.

The usual end-to-end evaluation compares different models based on their final outputs. This approach can be misleading in two directions. Models may produce the same final output even when their intermediate results differ, because surrounding code can correct, replace, or hide those differences. Conversely, models may produce different final outputs even when their intermediate results are similar, because a ranking or aggregation rule can amplify a small difference. Thus, a difference in the final output does not necessarily indicate a difference in model capability, and the same final output does not necessarily indicate similar model performance. To examine this problem directly, we audit all 15 decision points in REPURPOSEVL, a drug-repurposing agent pipeline we developed. At each decision point, we track which component supplies the value that is ultimately carried forward to the next step and contributes to the final result.

We compare four dense models from the same model family and tokenizer (27B, 9B, 4B, and 2B parameters). For each model, we evaluate the pipeline with a two-round, citation-grounded debate either enabled or disabled. This produces eight experimental conditions, with 25 replications for each condition. The two experimental factors are model size and debate, with common starting data, prompts, cache snapshots, and decision logic. The pathway cache subsequently evolves within each condition, so comparisons reflect both model behavior and the evolving pipeline state; they do not isolate a causal effect of model size.

Our findings demonstrate how final outputs can mislead. The 9B model produced divergent drug recommendations in two conditions, while the smaller 4B model matched the 27B baseline. A standard evaluation would label the 9B model as uniquely unstable. However, decision-point auditing revealed that the 9B deviations were sensitive to the final ranking rule, disappearing under four of the six aggregation policies we tested.

In this pipeline, fixed decision logic can hide process failures and amplify small differences in intermediate outcomes. Evaluating intermediate decision nodes alongside final outputs makes these effects visible.

**Contributions.** 1) In this setting, 4B gave the same aggregated recommendation with lower model-loading memory and model-call latency, but lower consistency and more citation rejection. 2) We trace decision paths: before debate, the model-path answer remained unchanged in 54% of matches, rule-based guards (K) wrote 22%, and deterministic deciders (F) wrote 24%. These are last-writer counts, not causal credit. 3) We recomputed the same 200 stored runs under six scoring rules without re-running the pipeline, separating tie-break-sensitive differences from one that survives every rule tested.

## 2 Related work

**Evaluation artifacts in benchmark scores.** Prior work notes that standard metrics often ignore operational costs (Kapoor et al. 2024), and oversimplify the multidimensional nature of performance (Zhu et al. 2026). Critically, the evaluation software itself can skew results more than the underlying model (Zhang et al. 2026; Yao et al. 2026) with the choice of parser alone moving a reported score from 0% to 76% on the same model responses (Garware and Zisad 2026). Recent taxonomies have begun to formalize the distinction between model failures and harness failures (Raj et al. 2026). We uniquely demonstrate that final benchmark metrics can erroneously report an aggregation tie-break as an intrinsic model property.

**Attribution and model substitution.** Prior work attempting to understand agent workflows relies on counting model calls, removing agents, assigning Shapley credit to tools, and tracing decision provenance (Zha et al. 2026; Lu et al. 2026; Horovicz 2025; Wang et al. 2026; Solozobov 2026). However, the credited module is often not the one performing the actual work (Cao et al. 2026). SLM research proposes small models for repetitive tasks, measures their tool-use limits, and allocates capacity by component, role, or query (Belcak et al. 2025; Karmakar and Chatterjee 2026; Srivastava et al. 2026; Jiang et al. 2026; Chen et al. 2025; Kotte 2026; Moslem et al. 2026; Pona et al. 2026; Huang et al. 2026). We instead trace which operational path supplies the value that survives at each decision point.

**Aggregation and strict outputs.** While multi-agent debate is generally shown to impact accuracy (Yang et al. 2025; Bertalanič and Fortuna 2026; Hao et al. 2026; Nilayam et al. 2026), rule-

based components can dominate noisy inputs in high-stakes decisions (Chun and Elkins 2026). Majority voting, social-choice rules, learned majority overrides, and correlated judge errors all shift the aggregated result (Choi et al. 2025; Rofin et al. 2022; He et al. 2026; Zhao et al. 2026). Additionally, prior work shows that strict output contracts often penalize weaker models (Fan 2026; Ray 2026; Galeone et al. 2026; Parikh 2026), harness sensitivity can be non-monotonic (Cho 2026); and influence tends to concentrate in a subset of agents (Bause et al. 2026). We add direct evidence that implementation defects can also affect model and serving configurations unequally (§7).

## 3 Setup

**Task.** The objective of the pipeline is to identify pairs of FDA-approved drugs that could be repurposed to treat idiopathic pulmonary fibrosis (IPF), a progressive disease characterized by scars in the lungs and has no cure and limited treatments (U.S. Food and Drug Administration 2025). The system's final recommendations serve as biological hypotheses for future experimental testing. The sixteen starting drug pairs come from an upstream gene-expression screen, are identical across all eight conditions, and are not generated by any tested model (Appendix A).

**Pipeline.** REPURPOSEVL utilizes a virtual-laboratory pattern (Swanson et al. 2025) where specialized agents exchange structured, cited claims rather than conversational text, creating a fully recorded decision train. Sixteen drug pairs enter safety screening, and twelve advance to an eleven-match, single-elimination tournament. Fifteen instrumented decision points, which we call the guard chain, can act on a match: nine guards that inspect and may correct the current winner (K-01–K-09) and six deterministic deciders (F-10–F-15). They are not a fixed sequence and not two separate paths: a decider can run inside a guard, and a point may execute more than once or not at all. From the semi-finals onward, a two-round debate among three reviewers and an adjudicator may overturn a verdict. A final aggregation step combines all 25 replications into one recommendation (Appendices B and C). The interface is shown in Appendix I (Figures 5 and 6).

**Instrumentation.** Every instrumented point is equipped with a logging wrapper that records its identifier, execution order, activation status, selected structured reads exposed to the logger, and observable state changes without changing the underlying process. For pre-debate provenance, we credit a point only when it fires, selects the pre-debate verdict (V0), and is not discarded inside a non-firing enclosing guard. If no K/F record survives these checks, the model-path answer is recorded as unchanged.

**Conditions.** We test four dense Qwen3.5-27B, -9B, -4B, and -2B models with debate on and off. Each model was served via vLLM with identical configurations and a 32,768-token context window (Appendix I). For each model, the debate-on and debate-off conditions ran concurrently on the same vLLM instance. The candidate fingerprint and five precomputed evidence caches are identical across all conditions and are re-copied before every replication. A preflight check ensured configuration fidelity (Appendix I). We ran each condition 25 times, with 11 matches per run: 200 runs and 2,200 matches in total.

The pathway-relevance cache, which stores judgments produced by the model, evolved during execution. Copied from a shared 446-entry snapshot, this cache updates dynamically and is shared across all replications within a condition, mirroring real-world deployment. Replications within a condition are therefore not strictly independent (Appendix I).

## 4 What the final output says

The question is not whether a small model works but at what size its behavior shifts, and comparing final outputs answers it oddly here. Every pair below shares testosterone-enanthate as its partner, so we name a pair by its other drug. Our results showed that five of the eight conditions recommend a colistin-based pair. Both 9B conditions and the 2B condition without debate recommend an ibrutinib-based pair. In this result, 4B matches 27B, 2B matches it in one of two conditions, and 9B completely disagrees (Figure 1; Appendix D reports every condition).

This end-to-end reading fails in two different ways.

**Failure 1: the boundary depends on the ranking rule.** When scaling down from the 27B baseline to the 9B model, the final drug recommendations diverge; however, scaling down further to the 4B model unexpectedly reverts the outputs back to the original 27B baseline. The underlying match records show no such reversal. Instead, this non-linear behavior is an artifact of the candidate sorting tuple: (novelty, finals reached, finals won, name). Because the novelty gate—which demotes drug pairs already well-covered in the literature—is tied among the relevant candidates (Appendix E), the decision falls to the second key. Consequently, the pipeline prioritizes the number of finals reached (whether won or lost) over the actual number of tournaments won, which is what produces the 9B deviation. The 4B reversion has a different source: colistin leads on both keys there, 21–20 in finals reached and 14–8 in wins.

In the 9B no-debate condition, the `ibrutinib` pair reached 24 finals and `colistin` reached 23, placing `ibrutinib` first. Yet `ibrutinib-based` pairs won only 5 tournaments, compared with 20 for `colistin-based ones`. A similar but narrower inversion occurred in the debate condition (ibrutinib-based pairs led 21–20 in finals reached but trailed 10–13 in wins).

We recomputed all recommendations from the stored run records under six ranking policies, changing only the aggregation rule. All four tested policies that prioritize tournament wins, including plain majority and Borda count, return `colistin-based pair` for both 9B conditions. Only the deployed rule and its novelty-free variant return `ibrutinib-based pairs` (Appendix D). In contrast, the 2B condition without debate returns `ibrutinib-based pairs` under all six policies; its run-level winners favor that candidate 14 to 11. The 9B deviations are therefore aggregation-sensitive, whereas the 2B deviation is robust to every policy we tested. A final-output comparison reports all three in the same way.

**Failure 2: the final output stays quiet while reliability worsens.** Between the 27B and 4B models, the final recommendation remains identical, and token usage shifts by only 4%. Median model-call latency falls by 13%, while model size and vLLM model-loading memory fall by about 7× and 6×, respectively. At 2B, token usage rises to 6% above 27B, which coincided with more ranking retries; the two figures come from different samples (three cost runs and 25 main runs; Figure 2, Appendix H). An evaluation that watches only final recommendations and aggregate token use sees a flat line followed by a step. This view fails to capture that internal reliability indicators were already deteriorating between 27B and 4B while the reported recommendation stayed the same.

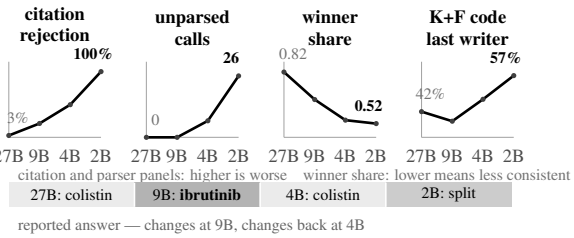



Figure 1: Internal traces change separately from the final recommendation. Citation rejection rises, unparsed debate calls appear, and the most-common-winner share falls. The containment-aware K+F share of pre-debate verdicts is 42.2%, 38.4%, 47.1%, and 56.5%; its complement is the model-path-unchanged share. Citation and parsing measures use debate-on runs; provenance pools both settings, and winner share averages their two 25-run shares. Last-writer provenance is not causal credit. The band shows final recommendations; split means the debate settings disagree.

## 5 What the internal traces say

For each match, the trace identifies the last writer of the pre-debate verdict V0. Its containment-aware rule credits an inner decider only if an enclosing guard adopts its value. Across 2,200 matches, the model-path answer remained unchanged in 54.0%, K guards wrote 22.0%, and F deciders wrote 24.0%; the largest loci were K-04 (13.5%) and F-11 (12.2%). These are not causal shares because K/F code can still depend on a model response. Figure 1 and Appendix B, Tables 3–4 report the trace measures and exact counts.

**Citation rejection rises sharply.** Our pipeline requires citations for substantive claims and verifies cited values against stored records. Here, grounding means a machine-verified citation, not factual or clinical correctness. Citation rejection is the fraction of schema-valid citations submitted to the verifier that fail verification. It rises from 2.8% at 27B to 21.1%, 49.7%, and 100.0% at 9B, 4B, and 2B (Figure 1). Unparsed responses, schema-invalid citations, and citations in `CONCUR` responses that bypass verification are excluded (Appendix B). The rule itself remains enforced: no citation-required claim is accepted without a valid citation. This reflects a failure of the smaller model to execute the task, not a relaxation of the pipeline rules. When measuring the fraction of runs won by the dominant candidate (averaged across debate conditions), consistency fell from 0.82 to 0.52. Crucially, a standard evaluation that looks only at the final, aggregated recommendation does not reveal this process degradation.

**The last-writer mix is non-monotonic.** The K+F share of pre-debate verdicts is 42.2%, 38.4%, 47.1%, and 56.5% from 27B to 2B (Figure 1, right panel). The model-path answer is the majority at 27B, 9B, and 4B; K+F is the majority only at 2B. This non-monotonic split locates the last surviving write under this guard configuration; it is neither a quality score nor causal credit.

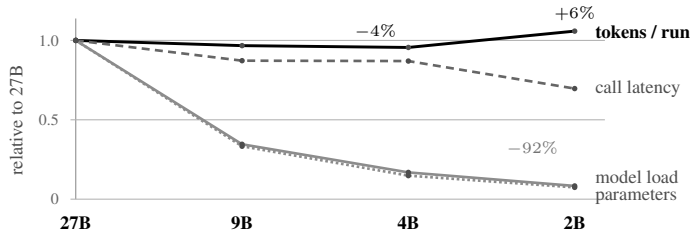



Figure 2: Resource measurements across model scales (debate disabled, three replications per condition). The recorded usage fields cover 100% of LLM calls; model-loading memory comes from the vLLM startup log. Across the full ladder, parameter count and vLLM model-loading memory fall by about 13× and 12×, respectively; median latency per model call falls by 30%. Token volume does not follow model size. Retry counts are measured on the 25-run main experiment, not on these three cost runs.

## 6 Why debate fails at 2B

The debate trace shows a distinct failure mode at the smallest size (Table 5, Appendix C). Despite producing 85 raw objections, the 2B model fails the citation contract: all 370 schema-valid citations submitted to the verifier are rejected, and 26 debate calls yield no parsed response. No objection retains a valid citation, so none advance to adjudication. All 75 debates receive the default `R1_CONCUR` status, which does not establish genuine reviewer concurrence. Thus, the citation requirement that organizes the 4B debate completely filters out the 2B model. This illustrates how macroscopic metrics conflate citation or parsing failures with genuine agreement, making a model whose objections are filtered out look like one with no objections. (Appendix G shows a well-formed 9B debate that does not alter the verdict).

The middle of the model-size range is also non-monotonic. The 9B and 4B models reach round two 61 times, but they overturn 2 and 16 verdicts, respectively. At the run level, at least one overturn occurs in 12 of 25 runs at 4B, compared with 2 of 25 at both 9B and 27B and 0 of 25 at 2B. Runs within a condition are dependent (Section 8). The observed 4B share exceeds 9B and 27B by 40 percentage points and 2B by 48 points. Nominal 95% Newcombe intervals under working independence are [+15, +59] and [+26, +67], respectively; these intervals have no guaranteed coverage for the actual dependent runs and are not used for significance claims. Reaching grounded debate and overturning a verdict are different abilities, and the debate layer is most active at 4B.

That activity rarely changes the reported recommendation. At 4B, debate produces 16 verdict overturns, yet colistin-based pair wins 13 of 25 debate-on runs and 14 of 25 debate-off runs. This one-run difference is -4 percentage points (nominal working-independence interval [-29, +22]). Because the two conditions follow different evolving cache trajectories, it cannot isolate a debate effect or establish that debate has no run-level effect. More generally, overturn counts do not track the raw difference between debate conditions: the largest such difference occurs at 9B (13 versus 20 colistin wins; -28 points, [-50, -2]) despite only two overturns.

**What aggregation does with these changes.** The two recommendations that differ at 9B are the aggregation-sensitive inversions in Section 4. Across all eight conditions, five are decided by at most one appearance on the active ranking key, and in two conditions the selected candidate won fewer tournaments than the runner-up (Appendix D). The final output therefore reflects the aggregation rule as well as the model and debate traces.

## 7 Evaluation software was biased before comparison

Evaluation software can be biased before any model is compared. Nine defects in ours passed the test suite and were corrected before the code revisions used for the reported comparison. Four had unequal effects across development model or serving configurations: two output-cap fixes addressed 9B request failures, while numeric/null and Boolean citation handling rejected valid 9B values. A separate container-citation defect was also documented in a preliminary 35B model; it was not specific to 9B. Three additional operational failures were silent: transport errors became no evidence, an endpoint could serve the wrong model, and failed search became no literature (Appendix F).

## 8 Limitations

The provenance shares are specific to this pipeline and guard configuration. Six aggregation policies reveal sensitivity to the deployed rule but are not exhaustive, and fixing nine defects does not make the harness defect-free. We also have no clinical ground truth: a changed verdict cannot establish which drug pair is medically better.

Provenance records the last writer, not counterfactual causal contribution; a K/F decision may still depend on model-generated evidence. In addition, the 25 runs in each condition share a pathway cache that each run updates and the next reads; it is the only state carried between runs (Appendix I), so the runs are not independent draws, and matches are clustered within runs. The intervals we report assume independent Bernoulli trials within each arm, an assumption these runs do not meet; we therefore treat them as working-independence intervals with no guaranteed coverage for the actual dependence structure, not as calibrated uncertainty estimates or significance tests. Confirmation requires resetting the cache per run or modeling serial dependence.

No energy was recorded for the final eight-condition comparison, so we cannot report success per watt. Cost results use three runs per condition and latency per model call, not end-to-end time. Models ran on separate H100 80GB GPUs of the same type; server-side latency may include GPU-instance variation. Reported memory is the vLLM model-loading footprint, not peak device memory or total working memory. It excludes KV-cache and runtime allocations. Neither this memory figure nor the latency measurement transfers directly to a phone or NPU. We also tested bfloat16, not on-device quantization.

The final records contain five pathway-verification timeouts in two runs (4B debate-on run 11 and 2B debate-off run 19). The affected calls return `NOT_SUPPORTED` on error. The runs are retained; because verification combines evidence across papers, these records alone do not establish whether the timeouts changed a final judgment or winner.

**Broader impact.** A drug recommendation could be mistaken for evidence of efficacy or safety and direct attention toward an unsuitable candidate. Trace auditing may reduce this risk by exposing fallbacks and ranking rules, but it cannot validate a clinical conclusion. We therefore present every output as a research hypothesis, not clinical advice.

Finally, the method needs decision points that can be named and wrapped. One point (`F-15`) was not exercised by the study inputs, so the study provides no empirical provenance evidence for that branch. The study covers one disease, tournament, candidate set, and model family; broader tasks, models, rules, and hardware require replication.

## 9 Conclusion

A macroscopic view of model scaling can be misleading. The 27B and 4B models produce the same aggregated recommendation, yet internal traces show higher citation rejection, unparsed de-

bate calls, and lower run consistency at 4B. Across model sizes, deterministic K/F code is the last surviving writer for 38–57% of pre-debate verdicts; the rest retain the model-path answer. Conversely, the changed outputs at 9B are sensitive to the aggregation rule: they disappear under four of the six policies we tested but persist under the deployed rule. The 2B debate-off deviation persists under all six tested policies. Thus matching outputs can hide a difference in process reliability, while differing outputs can reflect an aggregation tie-break rather than a model-only effect.

**What to measure instead.** Report the model-unchanged, K-guard, and F-decider shares, the most-common-winner share, and whether the recommendation changes under alternative aggregation. These diagnostics are not causal estimates, but they expose which path supplied the last surviving value and how sensitive the final metric is to the selected aggregation rule.

## References


Bause, Franka, Jonas Niederle, Martin Pawelczyk, and Rebekka Burkholz. 2026. "Multi-Agent Systems are Mixtures of Experts: Who Becomes an Influencer?" arXiv Preprint arXiv:2605.25929. https://arxiv.org/abs/2605.25929.

Belcak, Peter, Greg Heinrich, Shizhe Diao, Yonggan Fu, Xin Dong, Saurav Muralidharan, Yingyan Celine Lin, and Pavlo Molchanov. 2025. "Small Language Models are the Future of Agentic AI." arXiv Preprint arXiv:2506.02153. https://arxiv.org/abs/2506.02153.

Bertalanič, Blaž, and Carolina Fortuna. 2026. "The Cost of Consensus: Isolated Self-Correction Prevails Over Unguided Homogeneous Multi-Agent Debate." arXiv Preprint arXiv:2605.00914. https://arxiv.org/abs/2605.00914.

Cao, Xiaoyang, Siddarth Srinivasan, and Michiel A. Bakker. 2026. "Do Modules Stay in Their Lane? Role Drift in Compound LLM Systems." arXiv Preprint arXiv:2607.21627. https://arxiv.org/abs/2607.21627.

Chen, Lingjiao, Jared Quincy Davis, Boris Hanin, Peter Bailis, Matei Zaharia, James Zou, and Ion Stoica. 2025. "Optimizing Model Selection for Compound AI Systems." arXiv Preprint arXiv:2502.14815. https://arxiv.org/abs/2502.14815.

Cho, Yong-eun. 2026. "It's Not the Capability: Harness Sensitivity Is Non-Monotone Across LLM Agent Tiers." arXiv Preprint arXiv:2605.26731. https://arxiv.org/abs/2605.26731.

Choi, Hyeong Kyu, Xiaojin Zhu, and Sharon Li. 2025. "Debate or Vote: Which Yields Better Decisions in Multi-Agent Large Language Models?" arXiv Preprint arXiv:2508.17536. https://arxiv.org/abs/2508.17536.

Chun, Jon, and Katherine Elkins. 2026. "The Paradox of Robustness: Decoupling Rule-Based Logic from Affective Noise in High-Stakes Decision-Making." arXiv Preprint arXiv:2601.21439. https://arxiv.org/abs/2601.21439.

Fan, Hengxin. 2026. "Capacity, Not Format: Rethinking Structured Reasoning Failures." arXiv Preprint arXiv:2606.09410. https://arxiv.org/abs/2606.09410.

Galeone, Cosimo, Minsu Park, Giuseppe Ettorre, and Daniele Ligorio. 2026. "When Correct Isn't Usable: Improving Structured Output Reliability in Small Language Models." arXiv Preprint arXiv:2605.02363. https://arxiv.org/abs/2605.02363.

Garware, Chaitanya Vilas, and Sharif Noor Zisad. 2026. "When the Ruler is Broken: Parsing-Induced Suppression in LLM-Based Security Log Evaluation." arXiv Preprint arXiv:2605.07293. https://arxiv.org/abs/2605.07293.

Hao, Xiqi, Zengqing Wu, Yu-Xuan Qiu, Chuan Xiao, Ruiqi Xu, Shuyuan Zheng, and Jianbin Qin. 2026. "Not All Flips Are Conformity: Decomposing Stance Convergence in Multi-Agent LLM Debate." arXiv Preprint arXiv:2606.00820. https://arxiv.org/abs/2606.00820.

He, Chuan, Zebin Chen, Zhengyi Yang, Shaobo Qiao, Mingchen Ju, Jiate Liu, Dong Wen, and Guanfeng Liu. 2026. "Minority Sentinel: When to Overturn Majority Voting in Multi-Agent LLM Debates." arXiv Preprint arXiv:2606.29270. https://arxiv.org/abs/2606.29270.

Horovicz, Miriam. 2025. "AgentSHAP: Interpreting LLM Agent Tool Importance with Monte Carlo Shapley Value Estimation." arXiv Preprint arXiv:2512.12597. https://arxiv.org/abs/2512.12597.

Huang, Yiqian, Shiqi Zhang, Tianyuan Jin, and Xiaokui Xiao. 2026. "SCOPE: Cost-Efficient Model Selection for Compound AI Systems under Quality Constraints." arXiv Preprint arXiv:2606.00774. https://arxiv.org/abs/2606.00774.

Idiopathic Pulmonary Fibrosis Clinical Research Network. 2012. "Prednisone, Azathioprine, and N-Acetylcysteine for Pulmonary Fibrosis." *New England Journal of Medicine* 366 (21): 1968–77. https://doi.org/10.1056/NEJMoa1113354.

Jiang, Yinsicheng, Liang Cheng, Yeqi Huang, Yufan Zhao, Zhan Lu, Li Dong, Wenda Li, Edoardo Ponti, and Luo Mai. 2026. "Specialize Roles, Mix Deployments: Pushing the Cost-Accuracy Frontier of LLM Agent Teams." arXiv Preprint arXiv:2606.20629. https://arxiv.org/abs/2606.20629.

Kapoor, Sayash, Benedikt Stroebl, Zachary S. Siegel, Nitya Nadgir, and Arvind Narayanan. 2024. "AI Agents That Matter." arXiv Preprint arXiv:2407.01502. https://arxiv.org/abs/2407.01502.

Karmakar, Ranit, and Jayita Chatterjee. 2026. "AgentFloor: How Far Up the tool use Ladder Can Small Open-Weight Models Go?" arXiv Preprint arXiv:2605.00334. https://arxiv.org/abs/2605.00334.

Kotte, Varun. 2026. "UCCI: Calibrated Uncertainty for Cost-Optimal LLM Cascade Routing." arXiv Preprint arXiv:2605.18796. https://arxiv.org/abs/2605.18796.

Love, Michael I., Wolfgang Huber, and Simon Anders. 2014. "Moderated Estimation of Fold Change and Dispersion for RNA-Seq Data with DESeq2." *Genome Biology* 15 (12): 550. https://doi.org/10.1186/s13059-014-0550-8.

Lu, Mingyu, Yushan Huang, Chris Lin, and Su-In Lee. 2026. "Agents that Matter: Optimizing Multi-Agent LLMs via Removal-Based Attribution." arXiv Preprint arXiv:2605.27621. https://arxiv.org/abs/2605.27621.

Moslem, Yasmin, Magdalena Kacmajor, Vasudevan Nedumpozhimana, Ammar Abbas, Solmaz Panahi, David Lynch, Zhuangzhuang Nie, Alexandros Agapitos, Aleksandar Milenovic, Hongmeng Song, Yucheng Shi, Yue Pan, Patricia Buffini, and John D. Kelleher. 2026. "Cluster, Route, Escalate: Cascaded Framework for Cost-Aware LLM Serving." arXiv Preprint arXiv:2606.27457. https://arxiv.org/abs/2606.27457.

Nilayam, Prashanti, Kiran Ramanna, and Prashil Tumbade. 2026. "Detection Without Correction: A Two-Parameter Decomposition of Multi-Stage LLM Pipelines." arXiv Preprint arXiv:2605.27559. https://arxiv.org/abs/2605.27559.

Parikh, Tapan. 2026. "Structured Output Collapses Answer Diversity Across 44 Language Models." arXiv Preprint arXiv:2607.18476. https://arxiv.org/abs/2607.18476.

Pona, Edoardo, Milad Kazemi, Mehran Hosseini, Yali Du, David Watson, Osvaldo Simeone, and Nicola Paoletti. 2026. "Calibrate-Then-Delegate: Safety Monitoring with Risk and Budget Guarantees via Model Cascades." arXiv Preprint arXiv:2604.14251. https://arxiv.org/abs/2604.14251.

Raj, Harsh, Vipul Gupta, Anas Mahmoud, Razvan-Gabriel Dumitru, Darvin Yi, Aakash Sabharwal, and Yunzhong He. 2026. "Model or Harness? An Interaction-Centric Taxonomy for Localizing Agent Failures." arXiv Preprint arXiv:2607.28802. https://arxiv.org/abs/2607.28802.

Ray, Jaideep. 2026. "The Constraint Tax: Measuring Validity-Correctness Tradeoffs in Structured Outputs for Small Language Models." arXiv Preprint arXiv:2605.26128. https://arxiv.org/abs/2605.26128.

Rofin, Mark, Vladislav Mikhailov, Mikhail Florinskiy, Andrey Kravchenko, Elena Tutubalina, Tatiana Shavrina, Daniel Karabekyan, and Ekaterina Artemova. 2022. "Vote'n'Rank: Revision of Benchmarking with Social Choice Theory." arXiv Preprint arXiv:2210.05769. https://arxiv.org/abs/2210.05769.

Solozobov, Oleg. 2026. "Property-Level Reconstructability of Agent Decisions: An Anchor-Level Pilot Across Vendor SDK Adapter Regimes." arXiv Preprint arXiv:2605.12078. https://arxiv.org/abs/2605.12078.

Srivastava, Gaurav, Aafiya Hussain, Chi Wang, Yingyan Celine Lin, and Xuan Wang. 2026. "EffGen: Enabling Small Language Models as Capable Autonomous Agents." arXiv Preprint arXiv:2602.00887. https://arxiv.org/abs/2602.00887.

Subramanian, Aravind, Rajiv Narayan, Steven M. Corsello, David D. Peck, et al. 2017. "A Next Generation Connectivity Map: L1000 Platform and the First 1,000,000 Profiles." *Cell* 171 (6): 1437–1452.e17. https://doi.org/10.1016/j.cell.2017.10.049.

Swanson, Kyle, Wesley Wu, Nash L. Bulaong, John E. Pak, and James Zou. 2025. "The Virtual Lab of AI Agents Designs New SARS-CoV-2 Nanobodies." *Nature* 646 (8085): 716–23. https://doi.org/10.1038/s41586-025-09442-9.

U.S. Food and Drug Administration. 2014a. *Orphan Drug Designations and Approvals: Nintedanib*. Https://www.accessdata.fda.gov/scripts/opdlisting/oopd/detailedIndex.cfm?cfgridkey=344511.

U.S. Food and Drug Administration. 2014b. *Orphan Drug Designations and Approvals: Pirfenidone*. Https://www.accessdata.fda.gov/scripts/opdlisting/oopd/detailedIndex.cfm?cfgridkey=177903.

U.S. Food and Drug Administration. 2025. *FDA Approves Drug to Treat Idiopathic Pulmonary Fibrosis*. Https://www.fda.gov/drugs/news-events-human-drugs/fda-approves-drug-treat-idiopathic-pulmonary-fibrosis.

Wang, Yiqi, Jiaqi Zhang, Taotao Cai, Zirui Liu, Qingqiang Sun, Zequn Sun, Zhangkai Wu, Manqing Dong, Mingkai Zheng, Xuefei Yin, and Yanming Zhu. 2026. "From Agent Traces to Trust: A Survey of Evidence Tracing and Execution Provenance in LLM Agents." arXiv Preprint arXiv:2606.04990. https://arxiv.org/abs/2606.04990.

Yang, Yongjin, Euiin Yi, Jongwoo Ko, Kimin Lee, Zhijing Jin, and Se-Young Yun. 2025. "Revisiting Multi-Agent Debate as Test-Time Scaling: A Systematic Study of Conditional Effectiveness." arXiv Preprint arXiv:2505.22960. https://arxiv.org/abs/2505.22960.

Yao, Yilun, Xinyu Tan, Chao-Hsuan Liu, Yaoming Li, Zhengyang Wang, Wenhan Yu, Zhewen Tan, Yuxuan Tian, Guangxiang Zhao, Lin Sun, Xiangzheng Zhang, and Tong Yang. 2026. "Harness-Bench: Measuring Harness Effects across Models in Realistic Agent Workflows." arXiv Preprint arXiv:2605.27922. https://arxiv.org/abs/2605.27922.

Zha, Junli, Jinbo Wang, Chao Zhou, and Xiang Song. 2026. "Trace2Policy: From Expert Behavior Traces to Self-Evolving Decision Agents." arXiv Preprint arXiv:2606.10457. https://arxiv.org/abs/2606.10457.

Zhang, Yunbei, Janet Wang, Yingqiang Ge, Weijie Xu, Jihun Hamm, and Chandan K. Reddy. 2026. "Stop Comparing LLM Agents Without Disclosing the Harness." arXiv Preprint arXiv:2605.23950. https://arxiv.org/abs/2605.23950.

Zhao, Jitian, Changho Shin, Tzu-Heng Huang, Satya Sai Srinath Namburi GNVV, and Frederic Sala. 2026. "CARE: Confounder-Aware Aggregation for Reliable LLM Evaluation." arXiv Preprint arXiv:2603.00039. https://arxiv.org/abs/2603.00039.

Zhu, Pengyu, Li Sun, Philip S. Yu, and Sen Su. 2026. ""LLM Agent Performance" Is Not a Single Evaluation Target." arXiv Preprint arXiv:2602.03238. https://arxiv.org/abs/2602.03238.

# A The indication and the candidate pipeline

**Why IPF.** Idiopathic pulmonary fibrosis is a progressive, irreversible scarring of the tissue around the air sacs. It has no cure and only limited treatment options (U.S. Food and Drug Administration 2025). As of 2026, FDA-approved options include pirfenidone, nintedanib, and nerandomilast (U.S. Food and Drug Administration 2014b, 2014a, 2025). They are intended to slow loss of lung function, not to reverse established scarring. This leaves a clear research question for drug repurposing, while making every proposed pair a hypothesis rather than a therapy.

Three features make IPF a useful test case for REPURPOSEVL. First, two biological processes reinforce each other: injured lung-lining cells activate scar-forming cells, while pro-fibrotic macrophages sustain inflammation and scarring. The pipeline therefore evaluates pairs, with one drug aimed at each process. Second, every candidate arrives with structured evidence, including how strongly it reverses disease-related gene activity in each cell type, real-world exposure counts, adverse-event rates, and pathway annotations. Each numeric claim can therefore be linked to and checked against a stored record. Third, safety is a substantive filter: a prior IPF trial of broad immunosuppression was stopped early because the treatment group had more deaths and hospitalisations (Idiopathic Pulmonary Fibrosis Clinical Research Network 2012).

The candidate set is an input to the instrumentation, not a clinical claim. No combination reported in this paper should be read as a therapeutic recommendation; §8 explains why clinical effectiveness cannot be inferred here.

**Data.** Screening uses an integrated atlas of nine public IPF lung single-cell RNA-sequencing (scRNA-seq) cohorts: 932,826 cells from 189 donors and 12,486 genes. Batch correction reduces study-specific differences, and we harmonise all datasets to the same 18 cell-type labels (Table 1).

Table 1: Nine public IPF lung scRNA-seq cohorts. Cells counts the full integrated atlas; the donor columns count only donors entering the differential comparison. A further 7 SSc-ILD and 18 COPD donors (160,184 cells) remain in the atlas as context but are excluded from the IPF-versus-control comparison.

| Accession | Study | Cells | IPF | Control |
|---|---|---|---|---|
| GSE136831 | Adams et al. 2020, *Sci Adv* | 312,928 | 32 | 28 |
| GSE159354 | Gao et al. 2020, Cell Rep Med | 181,208 | 4 | 3 |
| GSE135893 | Habermann et al. 2020, *Sci Adv* | 89,326 | 12 | 10 |
| GSE132771 | Tsukui et al. 2020, Nat Commun | 81,221 | 3 | 3 |
| GSE122960 | Reyfman et al. 2019, *AJRCCM* | 70,372 | 5 | 8 |
| GSE146981 | Yao et al. 2021, AJRCCM | 62,969 | 11 | 9 |
| GSE128033 | Morse et al. 2019, *Eur Respir J* | 55,117 | 8 | 10 |
| GSE190889 | Heinzelmann et al. 2022, Eur Respir J | 53,617 | 3 | 3 |
| GSE157996 | Liang et al. 2022, J Clin Invest | 26,068 | 6 | 6 |
| Total | | 932,826 | 84 | 80 |

The screening contrast is IPF versus normal only: 84 IPF donors (433,356 cells) against 80 controls (339,286 cells). The IPF cohort matches the epidemiology of the disease — median age 67 (IQR 63–70), 65% male, 53% with a smoking history — while the control donors are younger (median 51), which is a property of the pooled public data rather than a design choice. For each cell type, gene counts are first summed within each donor, a procedure called pseudobulk aggregation, and IPF-versus-control differences are then estimated with DESeq2 (Love et al. 2014).

Drug signatures come from LINCS L1000 (Subramanian et al. 2017): 720,216 perturbation profiles over 34,418 compounds. Scores use 12,328 directly measured and computationally inferred genes rather than only the landmark-gene subset. Real-world exposure counts and adverse-event rates come from the electronic health records of a single academic medical centre (1.6M patients).

**How candidates are produced.** Eight of the eighteen cell types are assigned to two biological groups, or *niches*. The fibrotic niche follows the progression from injured alveolar cells through

transitional and basal cells to scar-forming myofibroblasts. The macrophage niche follows the immune-cell progression from monocytes to pro-fibrotic macrophages. A useful drug should reduce the disease-associated gene program in its target niche without suppressing the other niche. Cell types that are more likely to be damaged by the disease than to drive it are not used as targets.

Selection has two stages. The deterministic stage asks a simple question for every compound and cell type: does the drug push disease-altered genes back toward the control state? A signed Kolmogorov–Smirnov statistic summarises whether genes elevated in IPF tend to fall after treatment, and vice versa. An empirical null compares that score with scores from background compounds, making different queries more comparable. A drug qualifies only if it reverses the anchor cell types in one niche, has little effect on the other niche, passes toxicity and drug-class exclusions, and is FDA-approved. The remaining drugs are ranked by how broadly and strongly they reverse the disease signature, adjusted for real-world exposure. Drugs with no observed exposure enter the lowest tier because their safety rates have no empirical support.

The judgment stage reads this checked evidence package and proposes pairs across the two niches. It considers whether the drugs act through complementary mechanisms, whether the pair is novel relative to current care, whether its pathways fit IPF biology, and whether safety flags remain. It cannot recompute a score, override a failed check, or introduce a drug rejected by the deterministic stage. This division prevents model judgment from promoting a candidate that did not pass the numerical and safety checks.

**What the tournament receives.** The result is a fixed set of sixteen drug pairs, each with an evidence record for every relevant cell type. This is the candidate fingerprint in §3; it is identical in all eight conditions and checked before every replication. A model from a different family created the set once before the ablation. None of the four tested models creates its own candidates, so upstream selection is a fixed input rather than an experimental factor.

Deterministic code controls most upstream selection, whereas the model makes evidence-based judgments. In the downstream tournament, K/F code is the last surviving writer for 46% of pre-debate verdicts (§5), motivating decision-point provenance rather than crediting the whole output to one component.

## B Pipeline architecture

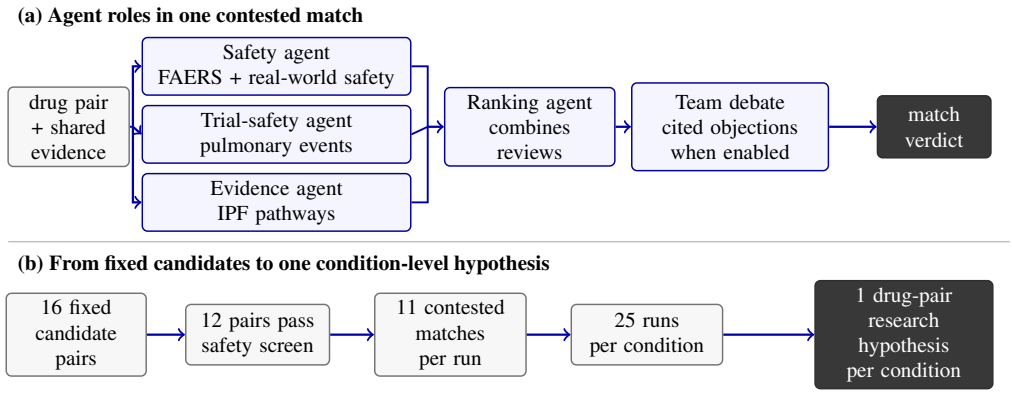



Figure 3: Simplified agent-level overview of REPURPOSEVL. The three reviewer boxes summarize role-specialized calls to the same condition checkpoint, not literal simultaneous execution: safety and clinical-trial reviews run first, and an early gate can stop a match before evidence review or ranking. The ranking call is validated and retried; if it still fails, deterministic fallback supplies a verdict, after which safeguards produce V0 before optional debate in enabled semi-final or final matches. Tables 2–4 describe those omitted controls. Each condition repeats the fixed 16-to-12 screen and a tournament of 11 contested matches 25 times, then combines the run records with a literature-derived novelty class under a fixed rule to produce one drug-pair hypothesis. The tested models did not generate the candidates, and logged provenance is observational rather than causal.

Figure 3 intentionally keeps the role-level flow simple. Tables 2–4 provide the conditional decision logic omitted from the overview. K-01–K-09 are conditional guards that may update the current verdict; F-10–F-15 are deterministic deciders over structured evidence. The IDs are not fixed steps: calls branch and nest, so an F may run inside K and not every point runs in every match. Table 2 lists their roles.

Table 2: What each of the fifteen instrumented points does when it is called. A point whose condition is unmet leaves the current winner unchanged, which is why several show high execution counts and zero fires in Table 4.

| Point | Role |
|---|---|
| `K-01` | Drop a candidate when it is the only one with both safety and clinical-trial evidence unclear; mechanistic evidence does not rescue it. |
| `K-02` | When pathway evidence is UNKNOWN or ABSTAIN, drop that axis and decide from safety, clinical trials, or the deterministic signal, rather than rewarding the other candidate for merely having data. |
| `K-03` | Re-check severe and critical safety grades after the model ranking and prefer the safer candidate. |
| `K-04` | Apply safety grade, then clinical grade. If neither decides, consult verified evidence and guarded reviewer-majority logic; the configured safety/evidence safeguard can suppress a majority override. |
| `K-05` | When a non-severe grade is tied and only one candidate holds positive EHR evidence, prefer that candidate. Suspended entirely if either axis is tied at severe or critical. |
| `K-06` | When safety and clinical trials are nondecisive, take the candidate with stronger machine-verified IPF evidence. |
| `K-07` | Compare pathway evidence only for the drugs the two combinations do not share, and only when safety and clinical trials are both tied. |
| `K-08` | A genotype-context tie-break inherited from the upstream pipeline. It has no IPF counterpart and is a pass-through here; its tie-break role is covered by K-06 and K-07. |
| `K-09` | Preserve partner-only and context-specific tie-breaks; otherwise reapply the broad verified-IPF-evidence comparison as a final consistency check. |
| `F-10` | If exactly one candidate carries a severe or critical safety grade, select the other. |

| Point | Role |
|---|---|
| F-11 | When safety and clinical grades cannot decide, compare the strength of machine-verified IPF evidence. |
| F-12 | Select the candidate holding a strict majority of non-abstaining reviewer votes. |
| F-13 | Compare the upstream seeding rank first; on a tie, use the combined score. |
| F-14 | When the safety and clinical reviewers select the same candidate and no suppression rule applies, take that candidate. |
| F-15 | If no earlier rule decides, default to candidate A of the bracket. |

Table 4 reports containment-aware provenance for the pre-debate verdict V0 across all 2,200 matches. A record is eligible only if it fires and selects V0. An eligible inner record is then discarded when it is contained by an enclosing guard that does not fire, because the inner value was evaluated but not adopted as that guard's write. The latest remaining eligible record receives credit; when none remains, the model-path answer is recorded as unchanged.

Under this rule, the model-path answer remains unchanged in 1,187 matches (54.0%), K-family guards write 485 (22.0%), and F-family deterministic deciders write 528 (24.0%). The largest individual loci are K-04 with 298 matches (13.5%) and F-11 with 269 (12.2%). Six points never receive V0 credit: K-01, K-03, K-08, K-09, F-12, and F-15. F-15 was not exercised by the study inputs; its wrapper is covered by unit tests but has no empirical count here.

This containment check matters empirically. A sequence-only rule that chooses the latest fired record disagrees with the precise rule in 943 of 2,200 matches, mainly because it credits an inner decider even when a non-firing outer guard does not adopt that write. As a cross-check, the precise rule reproduces all 300 stored v0_point labels from the debate traces.

Table 3: Trace-based measures across model sizes. Citation and parsing rows use debate-on runs; provenance pools both settings, and winner share averages their two run-level shares. The four provenance rows use the containment-aware pre-debate V0 rule: an inner decision is credited only if an enclosing guard adopts rather than discards it. K and F are reported separately, and their sum is the deterministic-code share. These changes are not reflected directly in Table 6, which reports the same recommendation at 27B and 4B.

| | 27B | 9B | 4B | 2B |
|---|---|---|---|---|
| Citations rejected on verification | 2.8% | 21.1% | 49.7% | 100.0% |
| Debate calls yielding no parsed response | 0 | 0 | 7 | 26 |
| Mean most-common-winner share | 0.82 | 0.66 | 0.54 | 0.52 |
| Pre-debate V0: model-path answer unchanged | 57.8% | 61.6% | 52.9% | 43.5% |
| K-family guard last writer | 13.5% | 20.9% | 26.2% | 27.6% |
| F-family deterministic decider last writer | 28.7% | 17.5% | 20.9% | 28.9% |
| All deterministic code (K+F) | 42.2% | 38.4% | 47.1% | 56.5% |

Citation rejection uses rejected/submitted counts of 4/141, 326/1,548, 820/1,649, and 370/370 for 27B, 9B, 4B, and 2B. Submitted citations partition exactly into accepted and rejected citations in all 300 debates. The denominator excludes 91 unverified citations in 9B CONCUR responses and excludes schema-stage drops and unparsed responses. Automatically generated fallback rulings reuse citations and are not counted as new verification events. The parsing row counts whole calls with no parsed response; it excludes partial item drops within otherwise parsed responses, including one 9B ruling.

Table 4: All fifteen instrumented decision points, pooled over eight conditions × 25 replications × 11 matches. "Executed" counts trace entries and "fired" records activation; neither alone implies provenance. "Decided" counts pre-debate V0 verdicts credited by the containment-aware last-surviving-writer rule. An inner fired record gets no credit when a non-firing enclosing guard does not adopt it. K-01–K-09 are guards that inspect and may correct the current winner; F-10–F-15 are deterministic deciders. The two families are not separate paths: an F point can run inside a K guard, and a K guard can inspect the value it receives. These are conditional points, so "executed" differs from 2,200 for most of them, and F-15 was never reached.

| Point | Class | Executed | Fired | Decided | Share |
|---|---|---|---|---|---|
| K-01 | guard | 4,400 | 0 | 0 | 0.0% |
| K-02 | guard | 1,491 | 4 | 1 | $< 0.1\%$ |
| K-03 | guard | 1,491 | 0 | 0 | 0.0% |
| K-04 | guard | 2,200 | 350 | 298 | 13.5% |
| K-05 | guard | 2,200 | 118 | 76 | 3.5% |
| K-06 | guard | 4,348 | 60 | 58 | 2.6% |
| K-07 | guard | 2,200 | 52 | 52 | 2.4% |
| K-08 | guard | 2,200 | 0 | 0 | 0.0% |
| K-09 | guard | 2,200 | 0 | 0 | 0.0% |
| F-10 | decider | 2,200 | 51 | 27 | 1.2% |
| F-11 | decider | 4,372 | 3,751 | 269 | 12.2% |
| F-12 | decider | 246 | 181 | 0 | 0.0% |
| F-13 | decider | 343 | 343 | 228 | 10.4% |
| F-14 | decider | 55 | 53 | 4 | 0.2% |
| F-15 | decider | 0 | 0 | 0 | 0.0% |
| No surviving K/F writer — model-path answer unchanged | | | | 1,187 | 54.0% |

## C Debate protocol

The debate opens once a match reaches the semi-final and proceeds in five steps. V0 is the verdict from the guard chain, together with its trace. Round 1 asks three reviewers, covering safety, clinical trials, and evidence integrity, to either `CONCUR` or `OBJECT`. An objection must name its type, propose a winner, and include citations. Adjudication accepts or rejects each objection and must also cite evidence. Round 2 sends every surviving objection back to all three review areas; each responds with `CONCEDE`, `MAINTAIN`, or `NEW_EVIDENCE`. V1 is the final verdict. If all reviewers concur in round one, V1 is identical to V0.

Every evidence-based claim must cite a stored source: a reviewer record, guard trace, or evidence packet. The citation also specifies the review area, drug-pair identifier, field, and claimed value. The validator compares that value with the stored record. A mismatched citation is removed individually, and a claim with no valid support is downgraded rather than accepted.

Table 5: Debate behaviour, 75 debates per model. Filed counts objections the reviewer produced; survived citation check counts those with at least one citation that resolved. The 2B files 85 and none survive: every debate terminates in round one because the validator discards its unsupported claims before adjudication.

| | 27B | 9B | 4B | 2B |
|---|---|---|---|---|
| Objections filed | 8 | 135 | 148 | 85 |
| survived citation check | 8 | 131 | 127 | 0 |
| Reached round two | 5 | 61 | 61 | 0 |
| Verdicts overturned | 2 | 2 | 16 | 0 |

## D Aggregation, and its two inversions

Table 6: Final recommendation per condition, 25 replications each. Five of eight agree. The common `testosterone-enanthate` partner is omitted; other pairs are written in full. The margin column is the finals-reached gap between the selected candidate and the runner-up; † marks the two conditions where the winner had fewer tournament wins than the runner-up. A zero means that finals reached tied and the later wins key decided, as in the two 2B conditions.

| Model | Debate | Per-replication winner | Final recommendation | Finals gap |
|---|---|---|---|---|
| 27B | on | colistin 20 / ibrutinib 4 / alvimopan 1 | colistin | 6 |
| 27B | off | colistin 21 / alvimopan 2 / ibrutinib 2 | colistin | 9 |
| 9B | on | colistin 13 / ibrutinib 10 / alvimopan 1 / bepridil–cetirizine 1 | ibrutinib† | 1 |
| 9B | off | colistin 20 / ibrutinib 5 | ibrutinib† | 1 |
| 4B | on | colistin 13 / alvimopan 6 / ibrutinib 5 / bepridil–colistin 1 | colistin | 11 |
| 4B | off | colistin 14 / ibrutinib 8 / bepridil–colistin 2 / alvimopan 1 | colistin | 1 |
| 2B | on | colistin 12 / ibrutinib 11 / cefoxitin–ibrutinib 2 | colistin | 0 |
| 2B | off | ibrutinib 14 / colistin 11 | ibrutinib | 0 |

Candidates are ranked by the tuple (novelty rank, finals reached, finals won, name). *Finals reached* counts appearances in a final whether won or lost, and is consulted before *finals won*. Across eight conditions this inverted the tournament's own verdict twice, decided a third by the same one-appearance margin, and decided the two 2B conditions on a tie.

| Condition | Candidate | Finals reached | Finals won | Ranked |
|---|---|---|---|---|
| 9B, debate on | ibrutinib | 21 | 10 | first |
| | colistin | 20 | 13 | second |
| 9B, debate off | ibrutinib | 24 | 5 | first |
| | colistin | 23 | 20 | second |
| 4B, debate off | colistin | 21 | 14 | first |
| | ibrutinib | 20 | 8 | second |

The 9B condition without debate is the clearest inversion: `colistin` wins 20 of 25 tournaments, compared with 5 for `ibrutinib`, but loses the recommendation because it reaches one fewer final. The two 9B conditions are the only conditions whose recommendation changes across the six tested aggregation policies (Table 7); the other six are policy-invariant. A final-output evaluation cannot distinguish this policy-sensitive difference from the policy-robust difference at 2B.

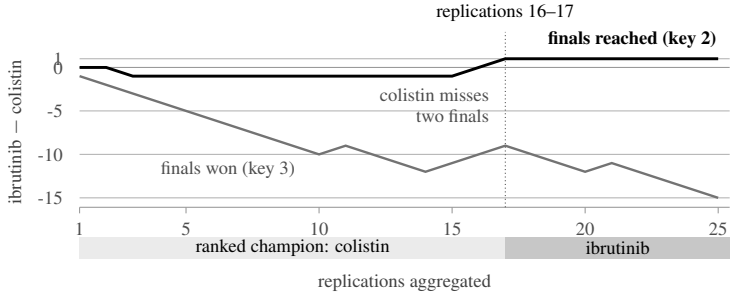



Figure 4: How the 9B recommendation without debate changes as runs are added. Each line is the cumulative gap, `ibrutinib` minus `colistin`, on one ranking key. Missing the finals in replications 16 and 17 moves `colistin` one behind on finals reached, even though it finishes 15 ahead in tournaments won. Because finals reached is checked first, the recommendation switches after replication 17 and later wins cannot change it.

Table 7: Final answer under six aggregation policies, recomputed from the same 200 replications with no re-execution. Only the two 9B conditions depend on the policy: they return `ibrutinib` under the two rules that read finals reached before finals won (P0, P5) and `colistin` under the four that read wins first. The other six conditions are policy-invariant — including 2B debate-off, whose `ibrutinib` survives every policy.

| Aggregation policy | 27B | 9B on | 9B off | 4B | 2B on | 2B off |
|---|---|---|---|---|---|---|
| P0 deployed: novelty > finals > wins | colistin | ibrutinib | ibrutinib | colistin | colistin | ibrutinib |
| P1 novelty > wins > finals | colistin | colistin | colistin | colistin | colistin | ibrutinib |
| P2 plain majority of wins | colistin | colistin | colistin | colistin | colistin | ibrutinib |
| P3 novelty > wins | colistin | colistin | colistin | colistin | colistin | ibrutinib |
| P4 Borda ($2 \times$ win + runner-up) | colistin | colistin | colistin | colistin | colistin | ibrutinib |
| P5 no novelty gate: finals > wins | colistin | ibrutinib | ibrutinib | colistin | colistin | ibrutinib |

These policies separate two deviations that a final-output evaluation reports in the same way. Both 9B recommendations are aggregation-sensitive: they change under every tested rule that ranks wins first. The 2B recommendation without debate is policy-robust: all six rules return `ibrutinib`, consistent with its 14-to-11 run-level win count. This comparison does not establish which drug is better or estimate a causal model effect; it shows whether the reported difference depends on the tested aggregation rule.

Figure 4 explains why the 9B result becomes fixed. After `colistin` misses the finals in replications 16 and 17, both candidates reach every remaining final. Their one-final gap therefore remains unchanged, and subsequent tournament wins cannot affect a ranking key that is never consulted.

## E The novelty-gate intervention

The gate demotes drugs already reported effective for the disease. Before the intervention it asked, in prose, whether an abstract showed the drug improving fibrosis. The two required enum fields we added are `drug_administered_alone` (boolean) and `fibrosis_change_attributed_to` with values {`drug_antifibrotic`, `resolution_of_other_pathology`, `not_measured`, `unclear`}. The code gate blocks an abstract whenever the first is false or the second is anything but `drug_antifibrotic`. Neither field tells the model what to conclude.

## F Harness defect catalogue

Nine defects were corrected before the code revisions used in the reported comparison; all had passed the test suite. Four had unequal effects across development configurations. An output cap set to the full context window caused HTTP 400 errors at 9B but not 27B; it was fixed at three call sites and then four remaining sites (two catalogue entries). Numeric/null citation handling and Boolean case normalization were two further defects: the development records show 10 of 97 9B citations affected by the former and 9 of 45 rejected 9B citations affected by the latter, with no such cases recorded for 27B. These percentages use different denominators and do not describe the final comparison.

The other five entries were an undocumented mandatory citation field, one illegal citation discarding a whole adjudication, a list-notation mismatch, two visible but uncitable fields (one entry), and container citation handling. The container fix explicitly records three 35B cases involving list/object values; it is not a 9B-only defect. These development observations concern particular prompts, validators, and serving configurations, not an intrinsic defect of a model size. Three further failures were silent: transport errors read as "no evidence", an endpoint quietly serving a different model, and a search failure indistinguishable from "no literature".

Six more we produced ourselves while re-running the experiments. Three were caught before publication: serving the 9B at half the stated context length; running a condition against the wrong seeding

input (43 candidate combinations instead of 16); and a GPU outage during which the runners retried a dead endpoint for three hours, so that a replication logged a champion after 57 connection errors and was recorded as normal — an outcome decided by the outage rather than by the guard chain. Three reached a draft: a determinism baseline that held at ten replications and failed at twenty-five; comparing two arms against pathway caches whose entries differed in 10% of positions, which produced the headline table of an earlier draft and inverted one condition (18 alvimopan-based wins became 20 colistin-based wins); and reporting two cells of Table 5 under different definitions, one counting objections filed and the other objections surviving citation check, in the direction that flattered the smaller model. Each was silent: no exception, no failing test, and in two cases a warm cache that made the run look ordinary.

## G An example debate

From the 9B, debate enabled, replication 1, match `R3_M2`: alvimopan against ibrutinib, both paired with testosterone-enanthate. The guard chain had ruled for ibrutinib. Elisions are ours; every citation shown resolved.

**Round 1: reviewer objections.** All three reviewers proposed alvimopan. Each returned `OBJECT` and labeled the objection `TIEBREAK_ORDER_CONFLICT`. The safety reviewer stated:

> The guard chain applied an EHR-positive tie-break (`K-05`) to select ibrutinib. . . but this contradicts the safety reviewer's explicit decision basis, where the decisive factor for both combos was `MANAGEABLE_NON_RESPIRATORY_TOXICITY`, rendering the EHR signal non-decisive.

Here *EHR-positive* denotes `K-05`'s asymmetric EHR-derived safety credit, not clinical efficacy; the cited enum records the basis for each candidate's grade, not the match winner.

> *Verified trace fields.* `reviewer/safety/...`: `grade_decisive_basis` for both pairs; `guard_trace/safety/...`: `K-05.after.winner`.

**Adjudication: three rejections.** In one ruling call, the adjudicator returned `REJECT` for all three objections. Its response to the safety objection stated:

> `K-05` explicitly fired an EHR-positive tie-break because the safety grades were tied at `MODERATE` and no `SEVERE/CRITICAL` escalation existed, which is the correct procedural step when the reviewer's non-decisive basis does not yield a winner.
>
> *Verified trace fields.* `guard_trace/safety/...`: `K-05.fired`, `K-05.after.winner`; `reviewer/safety/...`: `severity_grades`.

**Round 2: reconsideration.** The clinical-trials reviewer returned `CONCEDE`, stating that `K-05` had resolved a clinical-trials tie. That was the participant's interpretation; the trace records that `K-05` fired on tied `MODERATE` safety grades. The safety and evidence-integrity reviewers returned `MAINTAIN`. The latter claimed that `F-13` was the actual arbiter, but the containment-aware trace does not support that claim: `F-13` was an earlier decider, `K-04` subsequently selected alvimopan, and `K-05` then wrote the surviving V0 value, ibrutinib.

**Outcome.** The debate returned V1 equal to V0, so ibrutinib remained the final match winner. Every citation displayed above resolved. The case shows that a structured, cited exchange can leave the winner unchanged and can itself contain a mistaken provenance claim (§5).

## H Cost measurement

Wrappers around synchronous and asynchronous SDK completions record every LLM call; HTTP POST counts confirm 100% capture in all eight conditions. Each condition has three cost runs, sequential within a model and parallel across models on separate H100 80GB GPUs. A model's two debate settings share a server and run concurrently, so latency includes inter-arm contention.

Table 8: Token use and latency per replication, plus model-loading memory. With debate disabled, token volume changes by less than 5% from 27B through 4B, then rises: 2B uses 5.9% more tokens than 27B. Latency is median wall time per model call under normal pipeline concurrency. Memory is the vLLM startup log value for loading the model, not peak device memory. Neither measure is end-to-end runtime.

| | Tokens per replication | | Median call latency (s) | | Model load |
|---|---|---|---|---|---|
| Model | debate off | debate on | off | on | memory |
| 27B | 523,219 | 610,697 | 20.8 | 21.5 | 51.1 GiB |
| 9B | 505,929 | 637,890 | 18.1 | 17.3 | 17.7 GiB |
| 4B | 499,998 | 657,726 | 18.1 | 18.1 | 8.6 GiB |
| 2B | 553,875 | 609,912 | 14.5 | 15.9 | 4.3 GiB |
| *Debate premium*: 27B $+16.7\%$ 9B $+26.1\%$ 4B $+31.5\%$ 2B $+10.1\%$ | | | | | |

Token volume is not monotonic in model size. From 27B to 4B it falls 4.4%; at 2B it rises 5.9% above the 27B value. The smallest model triggers 460 ranking retries per 25 replications, compared with 341 for 27B; these counts come from the main experiment, whereas the token and latency figures above come from three cost-measurement runs per condition. In this server setup, going smaller reduces median call latency by 30% and model-loading memory by 92%, but does not reduce total processed tokens.

Debate activity and token premium are also non-monotonic. Relative to debate off, debate adds 17% more tokens at 27B, 26% at 9B, 32% at 4B, and 10% at 2B. The overturn counts (2, 2, 16, and 0) come from the 25-run main experiment, so the two series are not a cost-per-overturn ratio. At 2B, no debate reaches round two. These counts show resource use and protocol activity, not clinical or run-level benefit. Whether debate is worthwhile must therefore be assessed at each model size rather than inferred from the final recommendation alone.

# I Reproduction

Every model runs on an NVIDIA H100 80GB GPU under vLLM, with the arguments given below and nothing changed but the model name. The five baseline caches copied into every replication contain safety, clinical-trial, adverse-event severity, safety-verdict, and trial-cohort evidence. The separate 446-entry pathway cache is copied once from the common base at the start of each condition and then updated across its 25 replications. Each replication is a separate process, and the server ran with prefix caching disabled (hit rate 0.0% in the server logs), so this file is the only state carried from one run to the next. By the end of a condition its entries diverge by up to 38% from the starting snapshot and up to 40% between conditions. Changes include retrieved papers and queries, but also the stored judgment: up to 44/446 (9.9%) of judgment labels differ from the base and 57/446 (12.8%) between condition endpoints. These snapshot comparisons do not reconstruct the run-by-run trajectory or isolate the causal effect of cache state.

A preflight stops execution unless the served model identifier and context length match the declared configuration, the twelve combinations that survive screening match a reference fingerprint, the five baseline caches are present, and the declared pathway base exists. Each condition also stores `server_config.json` with the model, context length, port, debate setting, input paths, replication count, and code revision. A watchdog was added after an operational outage to stop runners after three consecutive failed server-health checks. The stored code revision does not capture uncommitted working-tree changes, so it does not establish watchdog coverage for every reported run. Configuration records also store cache paths rather than content hashes; future runs should record both source-tree and input-content hashes.

**Why dense models.** We excluded mixture-of-experts (MoE) models because their active computation does not scale linearly with total parameter count. For instance, a 35B MoE model activating only 8 of 256 experts per token might require less active computation than a 9B dense model. By evaluating only dense architectures, we ensure that the total parameter count across our four tested models provides a consistent, reliable measure of active model size and computational demand.

**Software and checkpoint versions.** The runtime is Python 3.10.12, vLLM 0.19.1, PyTorch 2.10.0, Transformers 5.5.4, and the OpenAI Python SDK 2.46.0; aggregation and interval com-

putation use NumPy 2.2.6, SciPy 1.14.0, statsmodels 0.14.6, and pandas 2.3.0. Weights are pinned to Hugging Face snapshot revisions: fc05daec for 27B, c2022362 for 9B, 851bf6e8 for 4B, and 15852e8c for 2B.

**Server launch command.** Every condition starts its server with the same command, changing only the model name. Port and host are omitted here because they are site-specific:

python3 -m vllm.entrypoints.openai.api_server --model Qwen/Qwen3.5-27B --served-model-name Qwen/Qwen3.5-27B --max-model-len 32768 --gpu-memory-utilization 0.90 --dtype bfloat16 --tensor-parallel-size 1 --max-num-seqs 16 --trust-remote-code --enable-auto-tool-choice --tool-call-parser hermes

**Decoding parameters.** Sampling temperature is fixed per agent: 0.1 for safety, clinical-trial, and ranking calls, 0.2 for debate calls, and 0.0 for the auxiliary safety-verdict call. The remaining sampling parameters, including top_p, use server defaults. The output cap is fixed in code at 4,096 tokens for every agent call. Although a preliminary ten-run measurement had a maximum output of 1,503 tokens, some recorded cost calls at 9B, 4B, and 2B reach 4,096 completion tokens; the cap is not universally inactive. Fixing it in code keeps the requested output limit identical across model sizes.

**Virtual-laboratory interface.** Nexus Virtual Lab (version 1) presents the pipeline through a web-based virtual-laboratory interface. A condition occupies one floor of a virtual institute, divided into six rooms that mirror the pipeline stages. The rooms display cited exchanges and values from stored records. This presentation adds no decision points and leaves the prompts, caches, and decision code of Section 3 unchanged. Figures 5 and 6 show representative views.

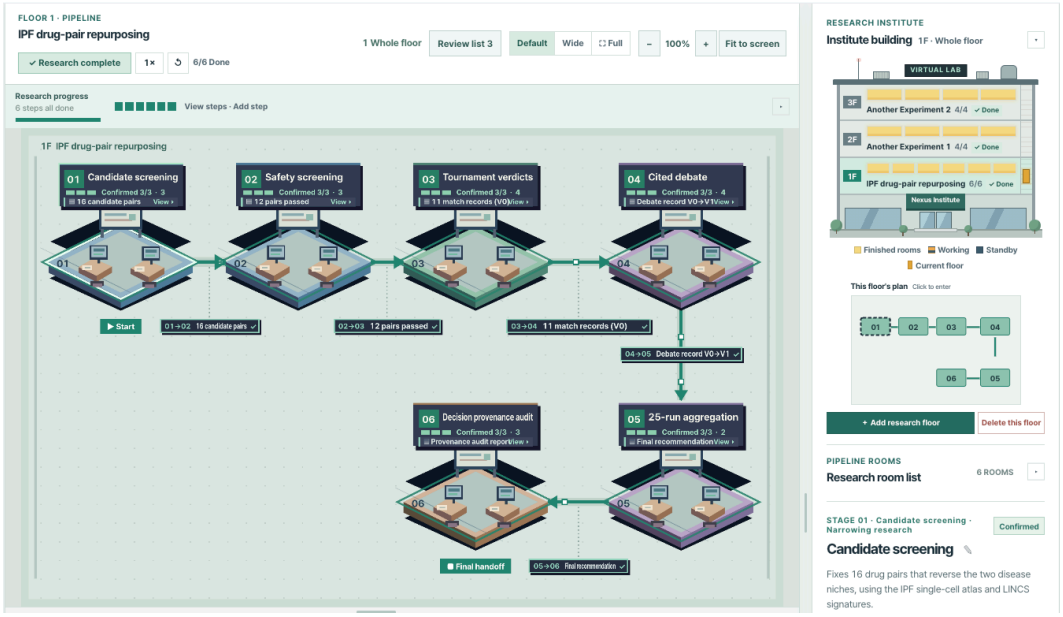



Figure 5: Pipeline view of one condition in Nexus Virtual Lab, version 1. The six rooms follow Section 3: candidate screening (16 pairs), safety screening (12 pass), tournament verdicts (11 match records, V0), cited debate (V0 to V1), 25-run aggregation, and decision provenance audit. Edges identify the records handed to the next room. The institute panel shows the floors and the selected floor's room plan.

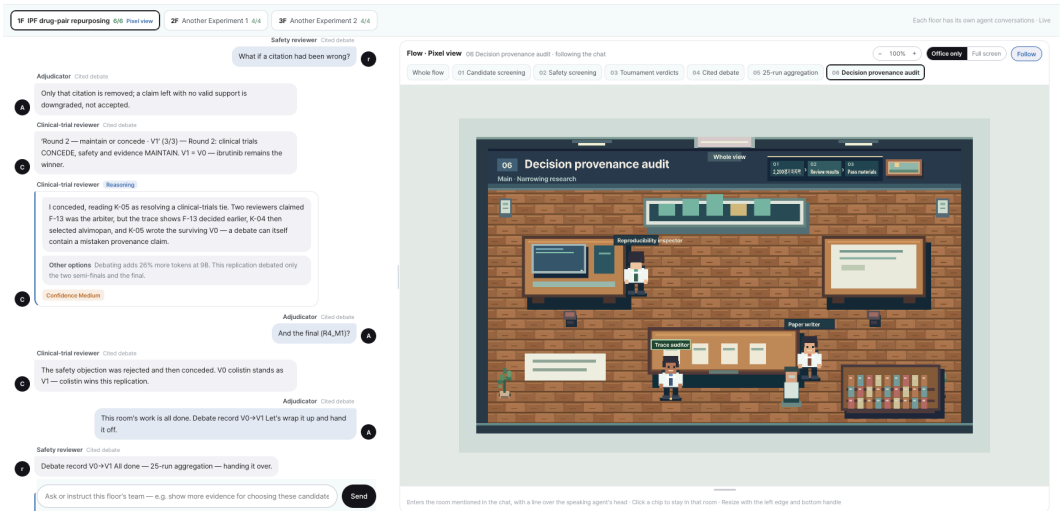



Figure 6: Decision provenance audit room (room 06) in Nexus Virtual Lab, version 1, with its three agents. The conversation panel (left) shows the cited exchange among the safety reviewer, clinical-trial reviewer, and adjudicator. The reasoning message restates the F-13 / K-04 / K-05 case analyzed in Appendix G.

**Randomization and seed handling.** The only explicit random path in the pipeline shuffles the seeding order (TOURNAMENT_RANDOMIZE_SEEDS), and it is off in all eight reported conditions. Run-to-run variation therefore comes from two sources: sampling above temperature zero, and the pathway cache that is updated across runs. Requests carry no per-request seed, so runs are not bit-for-bit reproducible; future runs should record request seeds as well.

**Code revisions.** The 27B and 9B conditions ran at 75d5f40, the 4B conditions at 4aaf103, and the 2B conditions at 8163217. Across these three revisions the differences in src/ are limited to two items: a token-usage recorder that is inert unless an environment variable is set, and an output cap in a translation script outside the pipeline. The decision-path code is identical in all three.

**Compute resources.** One NVIDIA H100 80GB HBM3 GPU serves one condition, with no tensor parallelism. The host has 124 vCPUs and 1.7 TiB of system memory, although the pipeline itself uses little CPU or storage. Median end-to-end time per replication with debate enabled is 14.7 minutes at 27B, 11.5 at 9B, 9.3 at 4B, and 10.0 at 2B. The 200 runs behind the main results took roughly 38 GPU-hours, and the 24 cost-measurement runs a further 4.5, for about 43 GPU-hours behind the reported results. Artifacts occupy 110–120 MB per condition, about 860 MB in total. Including preliminary and discarded runs, this study recorded 458 runs and roughly 202 GPU-hours.

## J Existing assets and licenses

This study releases no new model and no new dataset. The assets it uses are listed below with their creators, versions, and licenses or terms of use. Versions refer to the runtime described in Appendix I.

**Models and software.** The four tested models, Qwen3.5-27B, -9B, -4B, and -2B, are produced by the Alibaba Qwen team and distributed under the Apache License 2.0; their checkpoint revisions are given in Appendix I. Serving uses vLLM 0.19.1 (Apache-2.0), and weight loading uses PyTorch 2.10.0 (BSD-3-Clause) and Transformers 5.5.4 (Apache-2.0). The pipeline depends on the OpenAI Python SDK 2.46.0 (Apache-2.0), Qwen-Agent 0.0.34 (Apache-2.0), pydantic 2.12.5 (MIT), and requests 2.32.3 (Apache-2.0). Aggregation and interval computation use NumPy 2.2.6, SciPy 1.14.0, pandas 2.3.0, and statsmodels 0.14.6, all BSD-3-Clause. No model is fine-tuned and no weights are redistributed, so all use falls within the terms of these licenses.

**Data.** The nine IPF lung scRNA-seq cohorts in Table 1 are public NCBI GEO records, listed there with accession and source publication, and used under NCBI data-use policy. Drug signatures

come from the Broad Institute's LINCS L1000, distributed under CC BY 4.0. The pathway annotations carried in the upstream evidence are Gene Ontology terms, distributed by the Gene Ontology Consortium under CC BY 4.0. Adverse-event evidence comes from 53 quarters of FDA FAERS public files (2012Q4–2025Q4) and from the openFDA drug/event and drug/label endpoints, which are United States federal government works in the public domain and are used under the openFDA terms of use. Clinical-trial evidence comes from the ClinicalTrials.gov API v2, and literature evidence from NCBI E-utilities and the PMC Open Access subset, used under NLM and NCBI usage policies. Pseudobulk differential expression follows the DESeq2 method; that upstream step was run outside this repository, so its implementation version is not recorded here.

**Assets that cannot be redistributed.** The real-world exposure counts and adverse-event rates are aggregates derived from the electronic health records of 1.6 million patients at a single university hospital. They are bound by institutional terms of use and cannot be redistributed, which is why the paper cannot provide open access to data. The values reported are drug-level aggregates rather than individual patient records, and the source institution has been anonymized by the authors.

## NeurIPS Paper Checklist

1. Claims

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Section 1 state the contributions and observed scope, while Section 8 limits generalisation to one disease, pipeline, candidate set, model family, and hardware setup.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 8 discusses last-writer versus causal attribution, the evolving shared cache, dependence among runs, missing energy measurements, limited cost runs, hardware transfer, and the single-task and single-family scope.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper makes no theoretical claims and contains no theorems or proof results.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Section 3 and Appendix I give the design, the serving configuration, the exact vLLM launch command, the Hugging Face checkpoint revisions for all four models, package versions, per-agent decoding temperatures and the fixed 4,096-token output cap, seed handling, the code revision of each condition, and the compute budget. The instrumented decision points, the debate protocol, and the aggregation rules are specified in Appendices B, C, and D. Runs are not bit-for-bit reproducible because requests carry no per-request seed and the pathway cache is updated across runs; both are stated in Appendix I. Prompt texts and the EHR-derived inputs are not redistributable (Item 5). The reproducibility path is therefore procedural rather than artifact-based: the public inputs are identified by accession (Table 1), the guard chain, debate protocol and aggregation rules are specified in full (Appendices B-D), and the serving and decoding configuration is given exactly (Appendix I), so the study can be re-implemented by a group holding equivalent institutional exposure and adverse-event data. The reported numbers themselves cannot be regenerated without that data.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example

i. If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
ii. If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
iii. If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
iv. We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The paper lists the nine public cohort accessions in Table 1 and gives the full serving command, package versions, and decoding settings in Appendix I, but it does not release anonymized code or the processed data. The exposure counts and adverse-event rates consumed by the safety stage are derived from a single institution's electronic health records and cannot be redistributed under the institutional terms of use (Appendix J); the tournament cannot be run end to end without them.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: No model is trained; all settings are inference-time. Appendix I specifies package versions (Python 3.10.12, vLLM 0.19.1, PyTorch 2.10.0, Transformers 5.5.4, OpenAI SDK 2.46.0), the checkpoint revision of each model, the serving arguments, the per-agent sampling temperatures (0.0–0.2) with server-default top_p, the shared 4,096-token output cap, and how randomization is handled: the one explicit random path (seed-order shuffling) is off in all eight reported conditions, so run-to-run variation comes from sampling above temperature zero and from the pathway cache.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The central comparisons carry nominal 95% Newcombe intervals for the difference of two proportions, computed under working independence with the replication as the unit (n = 25 per condition) and reported in Section 6: the 4B overturn-run share exceeds 9B and 27B by +40%p [+15, +59] and 2B by +48%p [+26, +67]; the 9B debate-condition difference is -28%p [-50, -2]; and the same difference at 4B is -4%p [-29, +22], which includes zero under the working assumption. The method assumes independent Bernoulli trials within each arm, an assumption these runs do not meet because runs in a condition share an evolving cache (Section 8). They are therefore working-independence intervals with no guaranteed coverage for the actual dependence structure; observed proportion differences are descriptive, and no significance test is claimed.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix I reports one NVIDIA H100 80GB HBM3 GPU per condition with no tensor parallelism, the host CPU and memory, median end-to-end time per replication (14.7/11.5/9.3/10.0 minutes for 27B/9B/4B/2B with debate on), roughly 38 GPU-hours for the 200 main runs plus 4.5 GPU-hours for the 24 cost-measurement runs, about 860 MB of artifacts, and roughly 202 GPU-hours over the 458 recorded runs once preliminary and discarded runs are included. Appendix H reports model-loading memory and per-call latency.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: All authors have read and adhere to the NeurIPS Code of Ethics. The study analyses previously collected, de-identified data, releases no new model or dataset, and presents every drug pair as a research hypothesis rather than clinical advice.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The work may improve the reliability of SLM-agent evaluation, while an incorrectly attributed or clinically over-interpreted recommendation could cause harm. Sections 3, 8, and 9 and Appendix A discuss scope and mitigations.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release a new pretrained model, scraped dataset, or other high-risk model or data asset.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Appendix J credits every existing asset with its creator, version, and license or terms of use: the four Qwen3.5 checkpoints (Apache-2.0, revisions in Appendix I), vLLM, PyTorch, Transformers, the OpenAI SDK, Qwen-Agent and the analysis libraries; the nine public GEO cohorts, listed with accession and source publication in Table 1; LINCS L1000 and Gene Ontology terms (CC BY 4.0); FDA FAERS quarterly files (2012Q4–2025Q4) and the openFDA endpoints (public domain, openFDA terms); ClinicalTrials.gov API v2 and NCBI E-utilities/PMC (NLM and NCBI usage policies); and DESeq2. No model is fine-tuned and no weights are redistributed, so all use falls within the stated licenses. The EHR-derived aggregates are bound by institutional terms and cannot be redistributed, which is the reason for the No in Item 5.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The paper does not release a new dataset, model, or code artifact.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: This study utilized only pre-existing, fully deidentified data, which does not constitute human subjects research.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The study is a secondary analysis of previously collected, fully de-identified aggregate data and does not meet the definition of research with human subjects, so no IRB review or exemption determination applies to the work reported here. This is consistent with Item 14. No individual-level records were accessed by the authors of this analysis.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: Section 3 and Appendices B, C, and I describe the tested Qwen models, their agent roles and debate protocol, and the serving configuration because LLM use is central to the method.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.