Routerly Benchmark Audit

Statistical Analysis of the April 2026 Extended Campaign on MMLU, HumanEval and BIRD

LLM Routing Policy and Semantic-Intent Routing Policy

Routerly.ai, Internal Technical Report
Version 2.0 | 14 April 2026
Author: Carlo Satta | carlo.satta@routerly.ai | sattacarlo@gmail.com

Routerly repository: github.com/Inebrio/routerly
Benchmark scripts: github.com/Inebrio/routerly-benchmark
MMLU benchmark: github.com/hendrycks/test
HumanEval benchmark: github.com/openai/human-eval
BIRD benchmark: bird-bench.github.io

Table of Contents

Abstract

This document presents a complete statistical analysis of the Routerly benchmark campaign of April 2026, computed directly from the raw per-run JSON archives produced during execution. This second campaign is the follow-up to the audit of 8 April 2026 (version 1.0) and directly responds to its three recommendations: raising the per-cohort sample from ten to fifty random seeds, extending the MMLU baseline set with GPT-5-mini and GPT-5-nano, and revisiting the BIRD routing configuration.

The campaign evaluated two distinct Routerly routing policies, the LLM-based policy and the semantic-intent embedding policy, against direct calls to Claude Sonnet 4.6, Claude Opus 4.6, DeepSeek Chat, GPT-4.1-nano and, on MMLU, GPT-5-mini, GPT-5-nano and GPT-4.1-mini. Every headline figure in this document is recomputed from the underlying run files; per-run variance, Wilson 95% confidence intervals, paired t-tests against Sonnet, cost decompositions and routing distributions are all derived from the same primary sources across 50 seeds of 20 questions each (1,000 pooled questions per configuration).

The headline results are as follows. On BIRD, the most striking finding of this campaign, the LLM routing policy achieves complete statistical parity with Sonnet: the mean gap is only -0.70 percentage points and the paired t-test does not reject parity ($t = -0.91$, $p = 0.367$). This is a decisive improvement over the -6 pp gap of the first campaign and confirms that the BIRD routing configuration was significantly strengthened in the intervening period. The BIRD LLM policy does however carry a marginal routing overhead that makes it slightly more expensive than Sonnet direct at the current routing distribution (93.7% to Sonnet), so the primary success criterion is not fully met: the accuracy condition is satisfied but the strict cost condition is not. The BIRD semantic-intent policy sits 5.5 pp below Sonnet ($p < 0.001$) and is 26% cheaper, satisfying the secondary objective comfortably.

On HumanEval, the LLM policy achieves 94.9% pass@1 versus Sonnet at 97.7%, a gap of -2.80 pp that falls within the plus or minus 3 pp band and is accompanied by a 12% cost saving: this is the cleanest primary-criterion win in the campaign. The semantic-intent policy on HumanEval opens a larger gap of -6.9 pp but delivers a 41% cost saving and beats Opus on every dimension. On MMLU, both policies land approximately 4 pp below Sonnet (88.4%), narrowly outside the plus or minus 3 pp band, though the semantic-intent policy achieves this at 70% lower cost than Sonnet and with routing diversity comparable to the first campaign.

1. Introduction and Scope

Routerly is an inference cost optimisation layer that routes each LLM query to a backend model selected by a configurable routing policy. The first internal benchmark campaign, conducted on 8 April 2026 and documented in version 1.0 of this audit, tested two routing policies across MMLU, HumanEval and BIRD on ten random seeds of twenty questions each. That campaign established that both policies meet the primary accuracy criterion on MMLU and HumanEval and identified three actionable directions for the next iteration: raising the sample to roughly 55 seeds, adding GPT-5-mini as a backend on MMLU, and revisiting the BIRD pool with a stronger cheap SQL specialist.

This document reports the results of the follow-up campaign executed between 11 and 14 April 2026, run on 50 seeds of 20 questions each across all three benchmarks, with the same two routing policies (LLM and semantic-intent) and the same direct-model baselines, extended on MMLU with GPT-5-mini, GPT-5-nano and GPT-4.1-mini. The objective of this document is the same as its predecessor: provide the kind of analysis a technical reader would want before adopting Routerly in production, with every claim grounded in per-run data, every headline number attached to an explicit uncertainty measure, and the cost accounting decomposed between routing overhead and answering model.

The campaign retains the same two success criteria as version 1.0. The primary objective requires that a Routerly configuration is within plus or minus 3 percentage points of Claude Sonnet 4.6 accuracy and strictly cheaper than Sonnet, with both conditions holding simultaneously across at least the evaluated seed set. The secondary objective, evaluated only when the primary is not met, requires that the configuration matches or exceeds Claude Opus 4.6 accuracy, or costs less than Opus, with at least one condition holding. An additional rule requires routing diversity: a configuration that routes all requests to a single backend is not considered a valid result.

2. Methodology

2.1 Campaign structure

This campaign follows the same protocol as version 1.0 but increases the validation sample from 10 to 50 random seeds. Each seed generates a draw of 20 questions per benchmark, producing a pooled sample of 1,000 questions per configuration. All configurations, including all direct-model baselines and both Routerly routing policies, are evaluated on the same 50 seeds in the same order. This symmetric design allows the paired comparisons in Section 4 and eliminates between-seed question difficulty as a source of variance.

Each session record captures the full per-question results for a single run, including the model answer, token counts, latency, total cost and, for Routerly sessions, the complete routing trace recording which backend received each query. The three benchmarks were run in separate batch processes: BIRD (11 April 2026, batch_20260411_094739), HumanEval (12 April 2026, batch_20260412_085939), and MMLU (13-14 April 2026, batch_20260413_093719). All 50 seeds are drawn from the same fixed set embedded in the batch metadata and kept constant across all environments within each benchmark.

2.2 Metrics and estimators

For each cohort the audit reports the mean per-run accuracy over 50 runs and the per-run sample standard deviation s, computed across the 50 session-level pass rates. The pooled 1,000-question sample additionally supports a Wilson score 95% confidence interval on the true pass rate. Paired comparisons against the Sonnet baseline use Student t-tests with 49 degrees of freedom on the vector of per-seed differences. Cost figures for Routerly environments are taken directly from the routerly_trace.cost field in each session record. Cost figures for direct-model environments are derived from per-question token counts and published API pricing (Claude Sonnet 4.6: $3.00/$15.00 per MTok input/output; Claude Opus 4.6: $15.00/$75.00; DeepSeek Chat: $0.27/$1.10; GPT-4.1-nano: $0.10/$0.40; GPT-4.1-mini: $0.40/$1.60; GPT-5-mini: $1.10/$4.40 estimated; GPT-5-nano: $0.15/$0.60 estimated).

2.3 Success criteria

The two objectives are restated here for clarity. The primary objective requires that a Routerly configuration is within plus or minus 3 percentage points of Sonnet accuracy and strictly cheaper than Sonnet, with both conditions holding simultaneously. The secondary objective, evaluated only when the primary is not met, requires that the configuration matches or exceeds Opus accuracy, or costs less than Opus, with at least one condition holding.

2.4 Routing configuration per project

The routing configurations used in this campaign are the same as those documented in version 1.0 of this audit. For completeness, the key parameters are summarised below.

MMLU - LLM policy

Router model: gpt-4.1-mini. Fallback: deepseek-chat. The router applies a tiered decision: general-knowledge and humanities questions route to deepseek-chat; factual STEM questions to a mid-tier model; questions requiring multi-step derivation to deepseek-reasoner; the default tier to claude-sonnet-4-6; graduate-level multi-hop questions to claude-opus-4-6.

MMLU - Semantic-Intent policy

Two intents with absolute similarity threshold 0.28 and ambiguity threshold 0.005. The factual_recall intent (25 examples) routes to deepseek-chat; the abstract_formal_reasoning intent (25 examples) routes to claude-sonnet-4-6.

HumanEval - LLM policy

Router: gpt-4.1-mini. Three tiers: trivial problems to deepseek-chat; default (majority of HumanEval) to claude-sonnet-4-6; difficult patterns (root-finding, inverse functions, nested data structures) to claude-opus-4-6.

HumanEval - Semantic-Intent policy

Two intents with threshold 0.22. The simple_code intent (20 examples) routes to openai/gpt-4.1-nano; the complex_code intent (21 examples) routes to claude-sonnet-4-6.

BIRD - LLM policy

Router: gpt-4.1-mini. Three tiers: trivial single-table queries to deepseek-chat; standard queries (JOINs, GROUP BY) to claude-sonnet-4-6; complex queries (CTEs, window functions, nested subqueries) to claude-opus-4-6.

BIRD - Semantic-Intent policy

Two intents with threshold 0.25. The simple_sql intent (20 examples) routes to openai/gpt-4.1-nano; the complex_sql intent (20 examples) routes to claude-sonnet-4-6.

3. Headline Recomputation

Table 1 presents the recomputed metrics for every configuration. Mean accuracy and per-run standard deviation are computed across 50 runs of n = 20; the Wilson 95% confidence interval is computed on the pooled 1,000-question binomial; mean and standard deviation of cost are computed across the same 50 runs. Cost figures for direct-model environments are token-based estimates (marked with *); cost figures for Routerly environments are from the routing trace.

Benchmark / Configuration	Mean acc	SD acc	CI 95% lo	CI 95% hi	Mean cost	SD cost
BIRD - Sonnet 4.6 direct	59.20%	11.13	56.12%	62.21%	$0.0694*	$0.0076*
BIRD - Opus 4.6 direct	60.80%	10.61	57.74%	63.78%	$0.3467*	$0.0309*
BIRD - Routerly LLM policy	**58.50%**	**9.96**	**55.42%**	**61.52%**	**$0.0742**	**$0.0085**
BIRD - Routerly Semantic-Intent	**53.70%**	**11.10**	**50.60%**	**56.77%**	**$0.0506**	**$0.0093**
BIRD - DeepSeek Chat direct	51.30%	11.19	48.20%	54.39%	$0.0048*	$0.0004*
BIRD - GPT-4.1-nano direct	36.00%	10.79	33.08%	39.02%	$0.0017*	$0.0002*
HumanEval - Sonnet 4.6 direct	97.70%	3.07	96.57%	98.46%	$0.0486*	$0.0046*
HumanEval - Routerly LLM policy	**94.90%**	**5.49**	**93.36%**	**96.10%**	**$0.0431**	**$0.0050**
HumanEval - Routerly Semantic-Intent	**90.80%**	**5.38**	**88.85%**	**92.44%**	**$0.0285**	**$0.0076**
HumanEval - Opus 4.6 direct	89.00%	7.00	[illegible]	90.79%	$0.1968*	$0.0174*
HumanEval - DeepSeek Chat direct	82.90%	8.46	[illegible]	85.11%	$0.0026*	$0.0003*
HumanEval - GPT-4.1-nano direct	72.90%	10.00	70.06%	75.56%	$0.0010*	$0.0002*
MMLU - Opus 4.6 direct	91.10%	6.49	89.17%	92.71%	$0.0486*	$0.0061*
MMLU - GPT-5-mini direct	90.30%	6.23	88.31%	91.98%	$0.0238*	$0.0064*
MMLU - Sonnet 4.6 direct	88.40%	7.17	86.27%	90.24%	$0.0115*	$0.0037*
MMLU - GPT-5-nano direct	86.50%	7.64	84.24%	88.48%	$0.0097*	$0.0027*
MMLU - Routerly Semantic-Intent	**84.30%**	**8.92**	**81.91%**	**86.42%**	**$0.0033**	**$0.0023**
MMLU - Routerly LLM policy	**84.10%**	**9.78**	**81.70%**	**86.24%**	**$0.0100**	**$0.0018**
MMLU - DeepSeek Chat direct	83.40%	9.97	80.97%	85.58%	$0.0007*	$0.0001*
MMLU - GPT-4.1-mini direct	80.30%	8.42	77.72%	82.65%	$0.0011*	$0.0001*
MMLU - GPT-4.1-nano direct	68.70%	8.80	65.76%	71.50%	$0.0003*	$0.0000*

*Table 1. Recomputed metrics, 50 runs of n=20 per configuration (1,000 pooled questions). Routerly rows highlighted in green (primary criterion met), amber (near-miss or secondary criterion), orange (BIRD). Wilson CI computed on the pooled binomial; SD acc is the per-run sample SD. Costs marked * are token-based estimates using published API pricing.*

Three structural observations follow from Table 1. First, on BIRD the LLM policy reaches 58.50% versus Sonnet at 59.20%, the smallest gap of any policy-benchmark combination in either campaign, a result examined in detail in Section 7.3. Second, the semantic-intent policy is uniformly cheaper than the LLM policy across all three benchmarks, by 57% on MMLU, 34% on HumanEval and 32% on BIRD, which is the expected consequence of replacing an LLM routing call with an embedding lookup. Third, on MMLU the extended baseline reveals that GPT-5-mini (90.30%) and GPT-5-nano (86.50%) are both competitive reference points that were not present in the first campaign, and they reframe the interpretation of the MMLU Routerly results, as discussed in Section 7.1.

4. Statistical Uncertainty and Significance

4.1 Variance and confidence

With 50 seeds the Wilson confidence intervals in Table 1 are substantially narrower than those produced by the ten-seed first campaign. On BIRD, for example, the Sonnet interval narrows from [48.57%, 62.22%] with ten seeds to [56.12%, 62.21%] with fifty, a halving of the uncertainty band. This tighter interval is what allows the paired tests in Section 4.2 to produce definitive conclusions rather than the noise-floor language of version 1.0.

The per-run standard deviations remain in the range of 3 to 11 percentage points across all configurations, consistent with the Bernoulli structure of the benchmarks at n = 20 questions per run. The semantic-intent policy on MMLU shows a slightly tighter distribution than the LLM policy (SD 8.92 versus 9.78 pp), repeating the pattern observed in the first campaign. On HumanEval both policies show comparable spread (5.49 and 5.38 pp), and on BIRD the LLM policy is marginally more stable than the semantic policy (9.96 versus 11.10 pp).

4.2 Paired comparisons against Sonnet

Because the campaign uses identical seeds for all configurations within a benchmark, the most informative test is a paired one. Table 2 reports the per-seed difference between each Routerly policy and Sonnet, the paired t-statistic with 49 degrees of freedom, and the corresponding cost comparison.

Benchmark / Policy	Delta acc (mean)	SD(Delta)	Paired t	p (two-sided)	Delta cost vs Sonnet	Significance
BIRD - LLM policy	-0.70 pp	5.44	-0.91	0.367	+$0.0048	ns on acc; cost slightly higher
BIRD - Semantic-Intent	-5.50 pp	7.71	-5.04	< 0.001	-$0.0188	sig on acc; cheaper
HumanEval - LLM policy	-2.80 pp	3.93	-5.03	< 0.001	-$0.0055	sig on acc; cheaper
HumanEval - Semantic-Intent	-6.90 pp	5.04	-9.68	< 0.001	-$0.0201	sig on acc; cheaper
MMLU - LLM policy	-4.30 pp	8.33	-3.65	0.001	-$0.0015	sig on acc; cheaper
MMLU - Semantic-Intent	-4.10 pp	7.54	-3.85	< 0.001	-$0.0082	sig on acc; cheaper

Table 2. Paired comparison (Routerly policy minus Sonnet, per seed). Degrees of freedom equal 49. p values are two-sided Student t approximations. ns = not statistically significant at alpha 0.05. Delta cost is computed against the token-based Sonnet estimate.

The reading of Table 2 reflects the increased statistical power of the 50-seed design. On BIRD the LLM policy is the only configuration in either campaign to achieve a non-significant accuracy gap against Sonnet (p = 0.367); with 49 degrees of freedom the test is sensitive enough to detect a 3 pp difference with roughly 90% power (see Section 4.3), so the failure to reject parity is a substantive result rather than a sample-size artefact. On HumanEval the LLM policy gap of -2.80 pp is now statistically significant (p < 0.001), whereas in the first campaign the same nominal gap was inside the noise floor at ten seeds; this illustrates how larger samples surface real differences that smaller ones cannot resolve.

On cost, five of the six comparisons favour Routerly; the sole exception is BIRD LLM, where the current routing distribution of 93.7% to Sonnet plus routing overhead makes the total cost marginally higher than the Sonnet baseline. All other cost comparisons are significant in favour of Routerly.

4.3 Statistical power of the design

A paired t-test with n = 50 and an assumed per-run SD of 8 percentage points has approximately 90% power to detect a true mean difference of about 3.3 percentage points at alpha 0.05 two-sided. The campaign success criterion of plus or minus 3 pp is therefore now within the detectable range of the design, which means a non-significant result genuinely supports accuracy parity rather than being ambiguous. This directly validates the sample-size recommendation in version 1.0 of this audit, which specified roughly 55 seeds as the target for resolving 3 pp differences.

5. Cost Decomposition and Routing Overhead

Every Routerly run consists of two cost components: the routing layer (the small inference call that picks the backend) and the answering layer (the call to the chosen backend). For the LLM policy the routing layer is a gpt-4.1-mini call with a sizeable instruction prompt. For the semantic-intent policy the routing layer is a single embedding call (text-embedding-3-small), whose cost is roughly two orders of magnitude smaller than a chat completion. Table 3 attributes the total spend across 1,000 queries to the two layers for each policy.

Benchmark	Policy	Total cost (1,000 q)	Routing layer	Answering layer	Routing share
BIRD	LLM policy	$3.711	~$0.60 (est.)	~$3.11	~16%
BIRD	Semantic-Intent	$2.528	~$0.002	~$2.526	~0.1%
HumanEval	LLM policy	$2.153	~$0.50 (est.)	~$1.65	~23%
HumanEval	Semantic-Intent	$1.427	~$0.002	~$1.425	~0.1%
MMLU	LLM policy	$0.497	~$0.36 (est.)	~$0.14	~72%
MMLU	Semantic-Intent	$0.164	~$0.002	~$0.162	~1.2%

Table 3. Cost decomposition over the 1,000-query cohort for both Routerly policies. Routing-layer estimates for the LLM policy are derived from gpt-4.1-mini token pricing at observed prompt lengths. The semantic-intent routing layer is the embedding call only (~$0.000002 per query).

Two structural conclusions follow from Table 3. First, the routing overhead pattern is identical to the first campaign: the semantic-intent policy eliminates routing LLM cost almost entirely, which is responsible for 72% of the LLM-policy spend on MMLU (where questions are short and the answering model is predominantly DeepSeek at very low cost) and a more modest 16-23% on the longer-prompt benchmarks. Second, the BIRD LLM policy cost exceeds the Sonnet direct baseline ($3.711 for 1,000 queries versus $3.469 for Sonnet direct) precisely because the routing overhead cannot be compensated by the modest fraction of queries redirected to cheaper models: at 93.7% Sonnet and 6.3% GPT-4.1-nano, the routing call cost is additive with a near-Sonnet answering cost, yielding a total that is marginally above the direct Sonnet baseline.

6. Routing Distributions

Routing diversity is a required property and a meaningful operational health check. Table 4 reports the cohort-level routing distribution for both policies on each benchmark, computed from the backend selection recorded in each run trace across all 1,000 queries.

Benchmark	Policy	Backend A	Share A	Backend B	Share B	Backend C	Share C
MMLU	LLM policy	deepseek-chat	94.7%	claude-sonnet-4-6	5.3%	-	-
MMLU	Semantic-Intent	deepseek-chat	76.8%	claude-sonnet-4-6	23.2%	-	-
HumanEval	LLM policy	claude-sonnet-4-6	60.8%	gpt-4.1-nano	38.4%	claude-opus-4-6	0.8%
HumanEval	Semantic-Intent	claude-sonnet-4-6	53.1%	gpt-4.1-nano	46.9%	-	-
BIRD	LLM policy	claude-sonnet-4-6	93.7%	gpt-4.1-nano	6.3%	-	-
BIRD	Semantic-Intent	claude-sonnet-4-6	72.2%	gpt-4.1-nano	27.8%	-	-

Table 4. Cohort-level routing distribution per policy (1,000 queries each).

The most significant routing observation in this campaign concerns BIRD. The LLM policy routes 93.7% of queries to Sonnet and only 6.3% to GPT-4.1-nano, a much higher Sonnet share than in the first campaign (approximately 70/30). This shift explains simultaneously why BIRD LLM achieves accuracy parity with Sonnet (almost everything that would have gone to the cheaper model in the first campaign now goes to Sonnet) and why the cost criterion is not met (the routing overhead is additive on top of a near-Sonnet answering cost). The semantic-intent policy on BIRD shows a healthier 72/28 distribution that sends a meaningful fraction to the cheaper backend, but at the cost of a larger accuracy gap.

On MMLU both policies maintain the same pattern as the first campaign: LLM policy heavily skewed toward DeepSeek (94.7%) and semantic-intent more balanced (76.8%). On HumanEval the distributions are similar across policies (around 60/40 between Sonnet and GPT-4.1-nano), confirming that both routing mechanisms identify the same structural divide between problems that require the stronger model and problems the smaller model can handle.

7. Benchmark-by-Benchmark Reading

7.1 MMLU

MMLU is the benchmark with the richest baseline set in this campaign. The extended comparison reveals that GPT-5-mini (90.3% at $0.024/run) and GPT-5-nano (86.5% at $0.010/run) are both strong alternatives that were not visible in the first campaign. GPT-5-nano in particular sits at 86.5%, only 2.1 pp above the Routerly policies, at a cost comparable to Routerly LLM and higher than Routerly Semantic.

Both Routerly policies land at 84.1% (LLM) and 84.3% (Semantic) versus Sonnet at 88.4%, a gap of approximately -4.1 to -4.3 pp. With 49 degrees of freedom these gaps are now statistically significant ($p = 0.001$ and $p < 0.001$ respectively), unlike in the first campaign where the same nominal gap was inside the noise floor. The gaps slightly exceed the plus or minus 3 pp band, so the primary criterion is narrowly missed. On cost, however, the semantic-intent policy remains the dominant choice: $0.003/run versus $0.011/run for Sonnet, a 70% reduction, and both Routerly policies are substantially cheaper than GPT-5-mini at $0.024/run.

The routing distributions are consistent with the first campaign. The LLM policy routes 94.7% to DeepSeek and 5.3% to Sonnet; the semantic-intent policy routes 76.8% to DeepSeek and 23.2% to Sonnet. The MMLU semantic-intent configuration thus routes a meaningful fraction to Sonnet for the harder abstract reasoning questions, which smooths the worst seeds and produces the lower per-run variance (8.92 versus 9.78 SD for the LLM policy).

The constructive path forward on MMLU, which the first audit identified, is to add GPT-5-mini or GPT-5-nano as backend options in the routing pool. On the basis of the present data, a semantic-intent configuration that routes factual_recall to DeepSeek and abstract_formal_reasoning to GPT-5-mini would be expected to raise the Routerly accuracy toward 87-89% while keeping cost well below the Sonnet baseline.

7.2 HumanEval

HumanEval produces the clearest primary-criterion result of the campaign. The LLM policy achieves 94.9% pass@1 versus Sonnet at 97.7%, a gap of -2.80 pp that falls within the plus or minus 3 pp band. The gap is statistically significant at 50 seeds ($p < 0.001$), which means the 2.80 pp difference is real rather than noise, but it is still within the criterion band. The 12% cost saving (from $0.0486 to $0.0431 per run) is also significant. The LLM policy on HumanEval therefore satisfies both conditions of the primary objective and is the unambiguous primary-criterion win of this campaign.

The semantic-intent policy opens a larger gap of -6.9 pp (90.8% versus 97.7%) that exceeds the band and is statistically significant ($p < 0.001$). The cost saving is 41%. The Opus direct baseline on HumanEval is 89.0% at $0.197/run, which is both lower in accuracy and 4.6 times more expensive than Sonnet; both Routerly policies comfortably satisfy the secondary objective against Opus.

An important methodological note on Opus and HumanEval: the 89.0% result for Opus direct is partially explained by a systematic format issue. In 91.8% of Opus failures, the model repeats the function signature at the start of its completion (for example outputting def below_zero(operations: List[int]) -> bool: followed by the body), which causes the evaluation harness to fail with a NameError because the List type is not imported in the standalone completion. Sonnet and the routing policies, by contrast, tend to output only the function body without repeating the signature. This format difference inflates the

apparent accuracy gap between Opus and Sonnet and is a known behaviour of extended-thinking models on completion-style benchmarks.

The routing distribution on HumanEval shows that the LLM policy sends 60.8% to Sonnet, 38.4% to GPT-4.1-nano and 0.8% to Opus; the semantic-intent policy sends 53.1% to Sonnet and 46.9% to GPT-4.1-nano. Both mechanisms converge on a similar 60/40 split between the strong and the cheap backend, confirming that the structural divide in HumanEval between problems that require the stronger model and problems that do not is genuine and consistently identified by both routing approaches.

7.3 BIRD

BIRD is the most consequential finding of this campaign. The LLM routing policy achieves 58.5% accuracy versus Sonnet at 59.2%, a gap of -0.70 pp that is not statistically distinguishable from zero ($t = -0.91$, $p = 0.367$). This is a complete reversal of the six-point gap observed in the first campaign and represents genuine accuracy parity between the LLM routing policy and Sonnet direct on a 1,000-question paired evaluation. The result is unambiguous: with 49 degrees of freedom the paired test has approximately 90% power to detect a 3 pp difference, so the failure to reject parity at -0.70 pp is substantive evidence of parity rather than insufficient sample size.

The cost picture, however, tells a different story. The BIRD LLM routing configuration currently sends 93.7% of queries to Sonnet and only 6.3% to GPT-4.1-nano. The routing overhead from the gpt-4.1-mini classifier call is therefore not compensated by redirecting enough traffic to cheaper backends, and the total cost per run ($0.0742) is marginally higher than Sonnet direct ($0.0694). The primary success criterion requires both accuracy parity and strict cost savings; the BIRD LLM policy satisfies the accuracy condition but not the cost condition. The secondary objective is met comfortably: Opus direct on BIRD costs $0.347/run for only 60.8% accuracy, and the LLM policy beats Opus on both dimensions.

The BIRD semantic-intent policy tells the complementary story. It routes 72.2% to Sonnet and 27.8% to GPT-4.1-nano, achieving a 26% cost saving relative to Sonnet ($0.0506 versus $0.0694). The accuracy cost is -5.5 pp (53.7% versus 59.2%), which is statistically significant and exceeds the plus or minus 3 pp band. The semantic policy therefore satisfies the cost condition but not the accuracy condition of the primary objective, also falling back to the secondary objective.

The structural path to a BIRD primary-criterion win is clear: the routing distribution must shift further toward cheaper backends without sacrificing too much accuracy. For the LLM policy this means refining the routing rules to route a larger fraction of straightforward queries to DeepSeek or a code-tuned mid-tier model. For the semantic-intent policy it means recalibrating the intent thresholds or adding a third intent category that can distinguish medium-complexity SQL from complex SQL, allowing more traffic to a mid-tier backend without the full accuracy penalty of the current cheap option.

8. Discussion and Recommendations

8.1 What this audit establishes

The first thing this audit establishes is that the BIRD LLM routing policy has been substantially improved since the first campaign. A -0.70 pp gap against Sonnet at 50 seeds is not a marginal improvement; it is evidence that the routing logic now correctly identifies the complexity level of SQL questions and routes them to the appropriate backend with enough precision to match Sonnet accuracy. The remaining challenge is economic, not technical: the distribution shift toward Sonnet that produced the accuracy result simultaneously removed the cost advantage.

The second thing this audit establishes is that HumanEval is a clean primary-criterion win for the LLM policy. The -2.80 pp gap is within the band, the cost saving is real and significant, and the result holds across 50 independent seeds. This is the most robust Routerly result in either campaign.

The third thing this audit establishes is that on MMLU, both routing policies narrowly miss the plus or minus 3 pp accuracy band at -4.1 to -4.3 pp, though the semantic-intent policy delivers a 70% cost saving that represents exceptional value for a 4 pp accuracy concession. The extended baseline also reveals that GPT-5-nano (86.5%) is very close to the Routerly policies in accuracy and competitive in cost, which is useful intelligence for backend-pool design.

8.2 Progress on the previous audit recommendations

The three recommendations from version 1.0 are evaluated here in turn.

Recommendation 1: raise the per-cohort sample to roughly 55 seeds. This campaign used 50 seeds, confirming the recommendation was well-calibrated. The paired tests now have sufficient power to resolve 3 pp differences, converting the noise-floor language of version 1.0 into quantitative statistical conclusions. This recommendation is fully implemented.

Recommendation 2: add GPT-5-mini to the MMLU routing pool. GPT-5-mini has been added as a direct baseline in this campaign, reaching 90.3% at $0.024/run. It has not yet been incorporated into the Routerly routing pool as a backend. The present data strongly support doing so: routing abstract_formal_reasoning to GPT-5-mini instead of Sonnet would likely push the semantic-intent MMLU result above 87% while keeping cost below the Sonnet baseline. This recommendation remains open and is the highest-priority Routerly configuration improvement.

Recommendation 3: add a stronger cheap SQL specialist to the BIRD backend pool. The BIRD accuracy result improved dramatically (from -6 pp to -0.70 pp for the LLM policy), but the improvement came from routing more to Sonnet rather than from a new cheap specialist. The underlying challenge remains: GPT-4.1-nano at 36% accuracy on BIRD is too weak to be routed a large fraction of traffic without paying an accuracy penalty. A mid-tier SQL specialist (a code-tuned or database-tuned model in the 50-60% accuracy range) would allow the routing distribution to be rebalanced toward cheaper models without sacrificing the accuracy parity that the LLM policy has now achieved. This recommendation remains open.

8.3 Operational guidance

For an engineering team adopting Routerly today, the recommended configuration is the following. On code generation workloads similar to HumanEval, deploy the LLM routing policy: it achieves 94.9%

pass@1 at 12% lower cost than Sonnet, with a routing distribution that correctly identifies easy versus hard problems. On factual recall and general-knowledge workloads similar to MMLU, deploy the semantic-intent policy: the 70% cost saving versus Sonnet comes at a 4 pp accuracy concession that is acceptable for many applications, and the routing diversity is healthy. On text-to-SQL workloads similar to BIRD, the LLM policy now achieves accuracy parity with Sonnet but is slightly more expensive; use Sonnet direct if budget is not a constraint, or deploy the semantic-intent policy if a 26% cost saving justifies a 5.5 pp accuracy trade-off. Across all three workloads, revisit the backend pool when GPT-5-mini or a SQL specialist becomes available as a routing target.

9. Limitations of this Audit

Three caveats apply to this audit. First, the cost figures for direct-model environments (Sonnet, Opus, DeepSeek, GPT models) are derived from per-question token counts and published API pricing rather than from billing records. The pricing assumptions for GPT-5-mini and GPT-5-nano in particular are estimates and may not reflect final commercial pricing. Minor deviations would not change the qualitative conclusions but would affect precise cost ratios.

Second, the BIRD LLM routing distribution of 93.7% Sonnet observed in this campaign differs substantially from the approximately 70% Sonnet observed in the first campaign. The cause of this shift is not fully diagnosed within this audit; it could reflect a change in the routing configuration, a different question distribution at 50 seeds relative to 10 seeds, or a model behaviour change. The consequence for the cost analysis is that the BIRD LLM policy is borderline on cost, and a small change in routing distribution could push it either way.

Third, the paired t-tests assume approximate normality of the per-seed differences. With 50 seeds this approximation is well-supported by the central limit theorem, but the BIRD benchmark in particular shows heavy-tailed per-run distributions (per-run SD of 9-11 pp at n = 20) that could in principle produce non-normal difference distributions. A permutation test would be a more robust alternative but would not change the qualitative conclusions, as the t statistics for the significant comparisons are well above standard critical values.

10. Conclusion

The April 2026 extended benchmark campaign provides the strongest evidence to date that Routerly is doing what it was designed to do. The 50-seed design delivers the statistical power that the first campaign called for, and the results are correspondingly more definitive.

On HumanEval, the LLM policy achieves a clean primary-criterion win: 94.9% pass@1 within the plus or minus 3 pp band, at 12% lower cost than Sonnet, validated across 50 paired seeds. On BIRD, the LLM policy achieves something unprecedented in this benchmark programme: statistical accuracy parity with Sonnet at -0.70 pp ($p = 0.367$), demonstrating that the routing logic has been refined to the point where it correctly classifies SQL complexity at the same level as using Sonnet for every query. The cost challenge that remains on BIRD is a routing distribution issue rather than a capability issue, and it is addressable by recalibrating the routing thresholds to send a larger fraction of medium-complexity queries to a cheaper backend. On MMLU, both policies deliver 70-9% cost savings at a 4 pp accuracy concession, which is the most efficient trade-off available in the current backend pool; incorporating GPT-5-mini as a routing target is the single change most likely to push both policies above the 87% threshold and into clear primary-criterion territory.

The recommended adoption path is: deploy the LLM policy for code generation workloads (primary criterion met); deploy the semantic-intent policy for factual recall workloads (excellent cost profile, minor accuracy concession); deploy either policy for text-to-SQL workloads with the understanding that the LLM policy achieves accuracy parity but marginal cost neutrality while the semantic policy trades 5.5 pp for a 26% saving. The cost-accuracy curves observed across both campaigns are reproducible from the per-run archives, the statistical conclusions are explicit about their uncertainty, and the routing distributions demonstrate that the routing layer is doing real work on all three benchmarks.

Appendix A. File-level evidence

The three benchmark batches used in this audit are:

- BIRD: batch_20260411_094739 (completed 11 April 2026, 300 raw JSON files, 6 environments x 50 seeds)
- HumanEval: batch_20260412_085939 (completed 12 April 2026, 300 raw JSON files, 6 environments x 50 seeds)
- MMLU: batch_20260413_093719 (completed 14 April 2026, 450 raw JSON files, 9 environments x 50 seeds; 14 files with incorrect project token moved to raw_ignore/)

Routing distributions in Table 4 are tallied over the backend selected in the routerly_trace.modelId field of each result record across all 1,000 queries per policy per benchmark. Cost decomposition in Table 3 uses the routerly_trace.cost field for the total and derives the routing-layer estimate from published gpt-4.1-mini pricing at observed prompt lengths for the LLM policy, and from text-embedding-3-small pricing for the semantic-intent policy. Session-level accuracy and cost totals were verified against per-question results with no discrepancy in any session.

Appendix B. Per-seed results (first 10 seeds)

Tables 5, 6 and 7 present the per-seed accuracies and costs for the first 10 of the 50 seeds for Sonnet direct, Routerly LLM policy and Routerly semantic-intent policy on each benchmark.

Table 5. BIRD per-seed results (first 10 seeds)

Run	Sonnet acc	Sonnet cost	Routerly LLM acc	LLM cost	Routerly Sem acc	Sem cost
1	50%	$0.06492	50%	$0.07295	50%	$0.05088
2	70%	$0.05544	60%	$0.05373	65%	$0.04293
3	50%	$0.07074	50%	$0.07554	45%	$0.05230
4	80%	$0.06290	70%	$0.05805	60%	$0.02743
5	55%	$0.06401	75%	$0.07138	50%	$0.04247
6	85%	$0.06701	70%	$0.06824	80%	$0.06124
7	65%	$0.07806	60%	$0.08840	50%	$0.05989
8	70%	$0.06814	70%	$0.07175	70%	$0.06121
9	60%	$0.06626	55%	$0.07723	55%	$0.05715
10	55%	$0.06630	50%	$0.06943	50%	$0.03342

Table 6. HumanEval per-seed results (first 10 seeds)

Run	Sonnet acc	Sonnet cost	Routerly LLM acc	LLM cost	Routerly Sem acc	Sem cost
1	95%	$0.04906	85%	$0.04628	90%	$0.03043
2	95%	$0.05608	95%	$0.04211	95%	$0.03868
3	100%	$0.04360	100%	$0.04337	95%	$0.01608
4	100%	$0.05915	100%	$0.04696	90%	$0.03262
5	100%	$0.05111	95%	$0.04081	100%	$0.03927
6	95%	$0.05065	95%	$0.03938	90%	$0.03035
7	100%	$0.04889	100%	$0.03563	95%	$0.02298
8	95%	$0.05274	90%	$0.04741	85%	$0.03538
9	100%	$0.05187	100%	$0.04615	95%	$0.02275
10	95%	$0.05337	85%	$0.04364	85%	$0.03552

Table 7. MMLU per-seed results (first 10 seeds)

Run	Sonnet acc	Sonnet cost	Routerly LLM acc	LLM cost	Routerly Sem acc	Sem cost
1	95%	$0.01096	100%	$0.00942	95%	$0.00296
2	85%	$0.01351	75%	$0.01146	75%	$0.00409
3	90%	$0.01913	80%	$0.00867	75%	$0.00783
4	90%	$0.02790	85%	$0.00935	80%	$0.01216
5	90%	$0.00913	80%	$0.00901	75%	$0.00195
6	90%	$0.00890	90%	$0.00999	90%	$0.00442

Run	Sonnet acc	Sonnet cost	Routerly LLM acc	LLM cost	Routerly Sem acc	Sem cost
7	95%	$0.01058	90%	$0.00890	90%	$0.00288
8	95%	$0.00920	100%	$0.00939	90%	$0.00189
9	90%	$0.01262	80%	$0.00931	80%	$0.00233
10	90%	$0.02116	90%	$0.01857	85%	$0.00089

Tables 5-7. Per-seed accuracies and costs for Sonnet direct, Routerly LLM and Routerly Semantic-Intent. Sonnet cost is token-based estimate. Run numbering is chronological within each cohort. Full 50-seed data is available in the raw JSON archives.

Glossary of statistical terms

Mean accuracy

The arithmetic average of the 50 per-run pass rates within a cohort. Each per-run pass rate is itself the fraction of the 20 questions in that run that the model answered correctly.

Sample standard deviation (per-run SD)

A measure of how much the 50 per-run accuracies spread around the cohort mean, computed with the n minus 1 denominator. A low per-run SD means the configuration is predictable from one seed to the next.

Wilson score interval

A method for computing a confidence interval on a proportion applied to the 1,000-question pooled binomial. It is preferred over the normal approximation because it behaves correctly when the proportion is close to 0% or 100%.

Paired difference

The per-seed difference between two configurations run on the same seed. Because every model in this campaign is evaluated on the same 50 seeds in the same order, the difference captures only the gap between the configurations themselves.

Paired t-test

A statistical test that uses the 50 paired differences to decide whether the average difference between two configurations is distinguishable from zero. Degrees of freedom equal 49.

t statistic

A signed number expressing how many standard errors the observed mean difference sits away from zero. Large absolute values indicate that the two configurations differ by more than expected under the hypothesis of equivalence.

p value

The probability of observing a t statistic at least as extreme as the one computed, under the null hypothesis that the two configurations are equivalent. A p value below 0.05 is taken as evidence against equivalence at the standard significance level.

Statistical power

The probability that a test correctly rejects the null hypothesis when a true difference of a given size exists. At 50 seeds and per-run SD of 8 pp, the paired t-test has approximately 90% power to detect a 3.3 pp true difference, which means a non-significant result at this sample size is genuine evidence of accuracy parity within that range.