ONE SKILL, MANY MODELS: LEARNING REPRESENTATION COMPATIBILITY FROM SPARSE BEHAVIORAL TESTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Agent skills can transfer poorly across large language models (LLMs) even when their procedures are unchanged: models may respond differently to semantically equivalent choices in instruction structure and executable organization. We introduce **SKILLPROBE**, a budgeted framework that diagnoses these model-specific representation preferences before skill adaptation. **SKILLPROBEBANK** provides 156 matched probes and 360 semantically equivalent variants spanning 14 instruction and executable factors. **SKILLPROBE** uses a hierarchical Bayesian profile to track preference effects and uncertainty, then selects probes according to their expected improvement to representation decisions per unit cost. The resulting profile can condition existing methods for skill migration, generation, cold start, and version-drift repair. At 20% of the full-bank budget, **SKILLPROBE** achieves 84.3% agreement with full-bank representation decisions using 198K tokens, outperforming adaptive low-budget baselines at lower cost. Integrated with strong skill learners, it achieves the best average rank across ALFWorld, WebShop, SkillsBench, and SWE-Skills-Bench. Across seven additional transfer settings involving Gemma, Llama, Qwen, and a GPT-4o target, SkillGraph+**SKILLPROBE** outperforms SkillGraph in 27 of 28 paired benchmark comparisons. These results suggest that sparse behavioral diagnosis provides a reusable signal for adapting skills across model deployments.

1 INTRODUCTION

Large language model (LLM) agents increasingly reuse procedural knowledge as *skills*: structured packages of instructions, examples, scripts, tool interfaces, and checks. Early systems accumulated executable skill libraries, while recent approaches distill skills from experience, evolve skill graphs, or compile them for heterogeneous runtimes Wang et al. (2024); Xia et al. (2026); Li et al. (2026b); Chen et al. (2026). Yet benchmarks show that skill utility varies sharply by model and task, and that some curated or generated skills can even reduce performance Li et al. (2026a); Han et al. (2026). This creates a portability problem: a skill authored or optimized for one model may not transfer effectively to another.

Prior work shows that meaning-preserving changes to prompt format or demonstration order can substantially alter model behavior, and that preferred formats correlate only weakly across models Sclar et al. (2024); Lu et al. (2022). Agent skills amplify this sensitivity because their representations vary not only in wording but also in hierarchy, step granularity, control flow, contracts, validators, recovery logic, and code boundaries. Re-optimizing every skill for every target model is expensive; applying a global "best practice" ignores model-specific effects; and directly asking a model what it prefers does not provide behavioral evidence.

These limitations raise a fundamental empirical question: do different models exhibit stable and distinguishable preferences over semantically equivalent skill representations? We first investigate preference fingerprints across seven deployments: four Qwen3.5 scales, GPT-4o, Claude 3.7 Sonnet, and Gemini 2.5 Pro. Figure 1 presents three controlled observations. First, panel (a) shows that equivalent representations induce different performance across these LLMs. Second, panel (b) summarizes the differences as profile distances. Qwen3.5-9B is closer to Qwen3.5-35B-A3B than

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

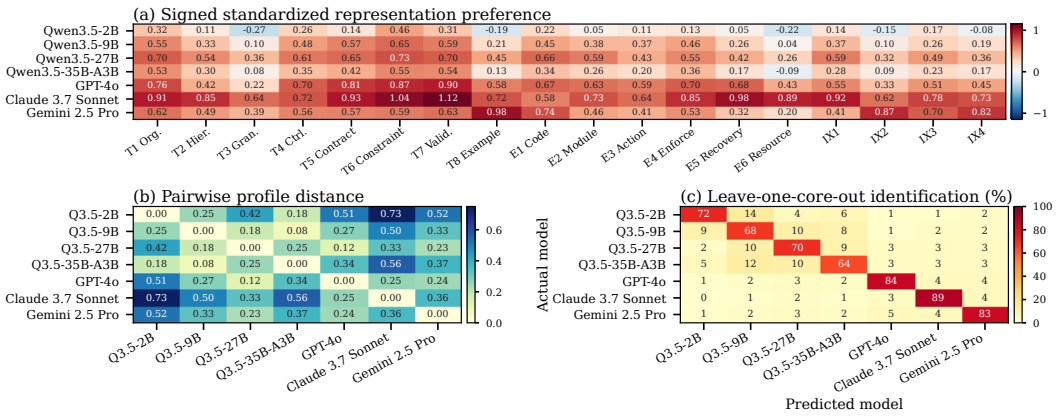


Figure 1: Evidence of model-specific skill representation preferences. (a) Signed standardized effects for eight instruction factors (T1–T8), six executable factors (E1–E6), and four interactions (IX1–IX4) across four Qwen3.5 scales, GPT-4o, Claude 3.7 Sonnet, and Gemini 2.5 Pro. (b) Pairwise distances between the resulting profiles. (c) Row-normalized model identification using held-out semantic cores. Each row corresponds to one actual model and sums to 100%; each cell reports the percentage identified as the corresponding predicted model. This is a profile diagnostic, not downstream task accuracy.

to Qwen3.5-27B, showing that representation preferences do not vary monotonically with parameter count. Third, the leave-one-semantic-core-out classifier in panel (c) achieves 75.7% accuracy, indicating that the profiles generalize beyond the cores used to estimate them. The matrix diagnoses model identity from held-out preference effects, not downstream task accuracy. Together, the panels reveal reproducible, model-specific representation fingerprints and motivate an efficient way to measure them before adapting skills.

We therefore propose **SKILLPROBE**, a low-budget probing framework that identifies which semantically equivalent skill representations work best for a target LLM. **SKILLPROBE** draws matched, semantically equivalent probes from **SKILLPROBEBANK** and adaptively evaluates only those expected to be most informative for downstream representation decisions. The resulting observations update a hierarchical Bayesian profile that captures representation preferences, uncertainty, and heterogeneous effects across task families. **SKILLPROBE** then converts this profile into structured evidence and an executable policy for deciding which aspects of a skill should be preserved, transformed, or deferred. Because the diagnosis is independent of any particular skill optimizer, it can be used as a plug-and-play conditioning signal for skill cold start, generation, migration, and version-drift repair. A single profile can further be amortized across a target model's skill library, avoiding repeated search for every model-skill pair. The main contributions are outlined below:

- We identify and formalize model-specific skill representation preferences as a distinct source of cross-model portability failure. We further release **SKILLPROBEBANK**, a dual-track benchmark that measures this phenomenon under controlled semantic equivalence and evaluates generalization on disjoint held-out semantic cores.
- We introduce **SKILLPROBE**, a budget-efficient diagnostic framework that formulates probing as Bayesian decision design. It infers an uncertainty-aware hierarchical preference profile and selects new or repeated probe blocks through cost-normalized robust decision-regret acquisition.
- We demonstrate that the resulting diagnosis is a reusable, plug-and-play conditioning signal for existing skill methods. At 20% budget, **SKILLPROBE** recovers full-bank decisions more accurately and cheaply than adaptive evaluation baselines, improves strong migration methods across four primary benchmarks, extends the migration evidence to Gemma and Llama families and a GPT-4o target, and also improves skill generation, cold start, version-drift repair, and search-augmented QA.

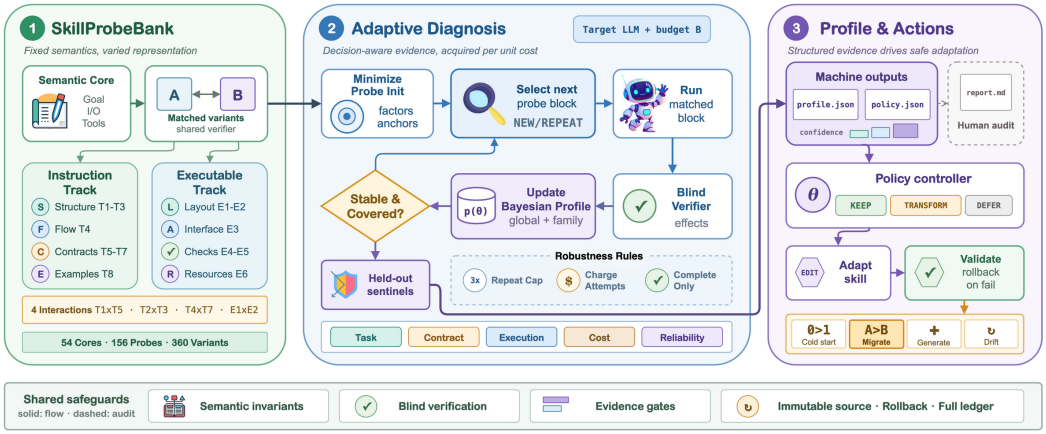


Figure 2: **SKILLPROBE** overview. Matched dual-track probes feed a cost-aware Bayesian NEW/REPEAT loop and held-out sentinel validation. Machine-readable profile/policy outputs independently condition migration, cold start, generation, and drift repair.

2 RELATED WORK

Agent skill learning, evaluation, and lifecycle. EvolveR, SkillRL, and SkillGraph distill reusable procedures or optimize skill libraries from experience Wu et al. (2025); Xia et al. (2026); Li et al. (2026b). SkillGen and EvoSkills synthesize skills with iterative verification Ma et al. (2026); Zhang et al. (2026). SkillsBench, SWE-Skills-Bench, and SkillLearnBench evaluate skill utility or continual generation Li et al. (2026a); Han et al. (2026); Zhong et al. (2026); recent evaluations further show that gains vary across agent–model configurations and weaken under realistic retrieval and refinement Shaposhnikov et al. (2026); Liu et al. (2026). SkillReducer and SkillOps address compression and library maintenance Gao et al. (2026); Pu et al. (2026).

Representation sensitivity and skill portability. Even semantically equivalent prompts can elicit substantially different behavior across models when their formatting or demonstration order varies Sclar et al. (2024); Lu et al. (2022). For skills, SkCC compiles an intermediate representation into framework-specific forms, and SkillRT uses model–harness capability profiles for portable execution Ouyang et al. (2026); Chen et al. (2026). Both establish the importance of portability, but neither estimates an uncertainty-aware preference over controlled representation contrasts under a low evaluation budget.

Prompt and skill optimization. DSPy/MIPRO, OPRO, and GEPA optimize instructions or language-model programs from task feedback Khattab et al. (2024); Opsahl-Ong et al. (2024); Yang et al. (2024); Agrawal et al. (2025). Such optimizers remain valuable downstream components, but they repeatedly spend validation budget for each model–skill pair. **SKILLPROBE** instead estimates an amortizable target-model profile that can condition the same optimizer or rewriter across a skill library.

Budgeted adaptive evaluation. tinyBenchmarks, Fluid Benchmarking, and continuous-score computerized adaptive testing reduce evaluation cost through item-response modeling or adaptive selection Polo et al. (2024); Hofmann et al. (2025); Balkir et al. (2026), building on Bayesian experimental design Chaloner & Verdinelli (1995). They primarily estimate aggregate ability, scores, or rankings.

3 PROBLEM FORMULATION

Setting. A *skill representation preference* is a deployment-conditioned performance ordering among realizations of a *semantic core* z, which fixes the goal, operations, branches, constraints,

Track	Instruction	Executable	**Total**
Families	6	4	**10**
Pool Cores	24	12	**36**
Sentinel Cores	12	6	**18**
Probes	114	42	**156**
Variants	264	96	**360**

Matched main-effect probe
`SPB-Q-I-ME-001`

core	`extract(full_name, email, team)`
factor	`T1: organization`
variants	`A: prose B: numbered_steps`
shared	`tasks T01--T04; exact-JSON verifier`
effect	$\text{score}(B) - \text{score}(A)$

Table 1: **SKILLPROBEBANK** coverage and matched probe example. Left: probe and variant counts refer only to the selectable pool. Right: a main-effect probe for the T1 organization factor.

validation, and recovery. Let $\mathcal{R}$ be the admissible representation space; $r \in \mathcal{R}$ realizes z as instructions, executable modules, or both. We write $r_i \sim_z r_j$ when two realizations preserve z and differ only on declared representation factors. Construction audits and shared verifiers establish this operational, not formal, equivalence. A deployment m fixes the model revision, agent scaffold, prompts, decoding, tools, permissions, and runtime. Complete paired outcomes from matched probe blocks form evidence D. Capability asks whether m solves z; preference compares performance under $r_i \sim_z r_j$.

Profile target. Let $\mathcal{Q}$ index declared representation contrasts, $\mathcal{K}$ the diagnostic channels, and $\mathcal{G}$ the task families. We define the target deployment profile as

$$\begin{aligned} \Phi_m(D) &= p(\Theta_m \mid D), \\ \Theta_m &= \{\theta_{q,k,g} \mid q \in \mathcal{Q},\ (k,g) \in \mathcal{K} \times \mathcal{G}\}, \end{aligned} \tag{1}$$

where $p(\Theta_m \mid D)$ is the posterior distribution and $\theta_{q,k,g}$ is the expected larger-is-better effect of contrast q on channel k in family g. Unlike an aggregate capability score, Φ_m retains effect direction, magnitude, uncertainty, and task-family heterogeneity. It belongs to the complete deployment identity m, rather than permanently to a base model. Section 5 instantiates this target.

Budgeted decision problem. At round t, a policy π selects a new matched block or interleaved repeat a_t at predeclared cost $C(a_t)$, yielding evidence D_T at its stopping time T. Let $h = (h_q)_{q \in \mathcal{Q}} \in \{-1, 0, +1\}^{|\mathcal{Q}|}$ encode the first contrast level, neutrality, or second level; unresolved evidence is deferred. Let $h^*(D_T)$ be the Bayes recommendation and $R(h, \Theta_m)$ its regret relative to an oracle that knows the true profile. Under budget B, we seek

$$\begin{aligned} \pi^* &= \arg\min_{\pi \in \Pi_B} \mathbb{E}_{\Theta_m, D_T \mid \pi}[R(h^*(D_T), \Theta_m)], \\ \Pi_B &= \left\{ \pi : \sum_{t=1}^{T} C(a_t) \le B \right\}. \end{aligned} \tag{2}$$

The expectation is over profile uncertainty, future probe outcomes, and policy randomness, making the objective decision-aware.

Full notation and protocol details are in Appendix A and Appendix C.

4 SKILLPROBEBANK

Design principles and factorization. **SKILLPROBEBANK** varies instruction presentation (T1–T8) and executable organization (E1–E6) while fixing each core's goal, I/O contract, tools, permissions, and failure behavior. The 14 factors, fixed before target-model evaluation, cover organization, hierarchy, granularity, control flow, contracts, constraints, validation, and examples on the instruction track; and code externalization, decomposition, interfaces, validation, recovery, and resources on the executable track. Dedicated 2×2 probes test the predeclared T1×T5, T2×T3, T4×T7, and E1×E2 interactions; other interactions are exploratory and strongly shrunk. Factor levels and invariants are in Appendix B.

Probe object and example. A probe is a matched core with two variants (four for an interaction), shared fixtures and verifier, a target contrast, and frozen cost. Let $\mathcal{P}$ denote the selectable probes.

Only organization changes, so the paired effect measures T1 in Table 1(right) preference rather than extraction ability.

Coverage and splits. Table 1 shows the 2:1 split of 54 cores across six instruction and four executable task families. Every family occurs in both splits, with core-derived artifacts kept together. Each core has one smoke and four scored fixtures (270 tasks total). The pool supports selection and fitting; disjoint sentinels test generalization and drift only after diagnosis. Downstream transfer uses separate public benchmarks.

Equivalence and validation. Variants may change only the declared factor while preserving each core's semantic units. Admission requires schema and reference checks, unit coverage, a factor-leakage audit, shared fixtures, and two independent raters; disagreements are adjudicated against the core. Deterministic verifiers are blinded to factor labels and expected directions. The Bank is frozen before target-model evaluation with immutable identifiers. Appendix B.3–B.4 details equivalence, length control, scoring, and release schemas.

5 SKILLPROBE METHOD

Figure 2 summarizes diagnosis and profile-conditioned applications.

Probe execution and effect extraction. Each run yields separate larger-is-better task, contract, execution, token-efficiency, latency-efficiency, and reliability utilities ($k \in \mathcal{K}$), without collapsing them into one score. Let o index a complete matched fixture–repeat comparison, with outcomes $Y_{o,\ell}$ for two levels and $Y_{o,ij}$ for a 2×2 block. We extract

$$\begin{aligned} \Delta^{\text{main}}_{o,k} &= u_k(Y_{o,1}) - u_k(Y_{o,0}), \\ \Delta^{\text{int}}_{o,k} &= \sum_{i,j\in\{0,1\}} (-1)^{i+j} u_k(Y_{o,ij}), \end{aligned} \tag{3}$$

where u_k is frozen and the second line is difference-in-differences. Positive effects favor the second level or joint use. Task quality drives decisions; other channels expose validity, reliability, and efficiency tradeoffs. Observation maps and failure treatment are in Appendix A.1.

Hierarchical Bayesian profile. Let $q(o)$ and $g(o)$ identify the contrast and task family of $\Delta_{o,k}$. We model a global effect plus a contrast-specific family deviation:

$$\begin{aligned} \theta_{q,k,g} &= \mu_{q,k} + b_{q,k,g}, \\ b_{q,k,g} &\sim \mathcal{N}(0, \tau^2_{q,k}/\kappa_f). \end{aligned} \tag{4}$$

The observed effects follow

$$\Delta_{o,k} \sim \mathcal{N}\Big(\theta_{q(o),k,g(o)}, \sigma^2_{g(o),k}\Big). \tag{5}$$

Here $\sigma^2_{g,k}$ is observation noise, $\tau_{q,k}$ captures between-family variation, and κ_f controls shrinkage. Channels are fit independently; opposing confident family effects produce `heterogeneous` rather than a global preference. Priors and outputs are in Appendix C.2.

Decision-aware profile readout. For contrast q, $h_q \in \mathcal{H} = \{-1, 0, +1\}$ selects its first level, neutrality, or second level. With normalized family and contrast weights ν_g, w_q, define $\vartheta_q = \sum_g \nu_g \theta_{q,\text{task},g}$ and $e_q = \operatorname{sgn}(\vartheta_q)(|\vartheta_q| - \rho_q)_+$ for task-quality ROPE ρ_q. We minimize oracle-relative regret:

$$\begin{aligned} R(h, \Theta_m) &= \sum_{q\in\mathcal{Q}} w_q \left[\max_{\tilde{h}_q \in \mathcal{H}} \tilde{h}_q e_q - h_q e_q \right], \\ h^*(D) &= \arg\min_{h\in\mathcal{H}^{|\mathcal{Q}|}} \mathbb{E}_{\Theta_m|D} R(h, \Theta_m). \end{aligned} \tag{6}$$

Neutrality costs $|e_q|$ when one level is better; choosing the opposite level costs $2|e_q|$. DEFER is post-decision abstention when confidence, coverage, or heterogeneity gates reject $h^*(D)$. Thresholds are in Appendix C.2.

Algorithm 1 Budgeted Skill Representation Diagnosis

Require: Bank $\mathcal{P}$, deployment m, budget B, thresholds (ϵ_R, ϵ_A)
Execute budget-feasible D-optimal S_0; retain complete effects D; fit $p(\Theta_m \mid D)$
$b \leftarrow B - C(S_0)$; $t \leftarrow 0$; $c_R \leftarrow \mathbb{I}[\mathcal{R}(D) \le \epsilon_R]$
while $\mathcal{A}_t = \{a \in \mathcal{P} : C(a) \le b,\ n_t(a) < 3\} \neq \emptyset$ **do**
  if $\Gamma(D) = 1$ and $c_R \ge 2$ **then break**
  Score each $a \in \mathcal{A}_t$ by Eq. (7)
  if $\Gamma(D) = 1$ and $\max_a \text{score}_t(a) < \epsilon_A$ **then break**
  Execute a_t^*; $b \leftarrow b - C(a_t^*)$; if complete, append effects and refit profile
  If updated, set $c_R \leftarrow c_R + 1$ when $\mathcal{R}(D) \le \epsilon_R$, else 0; $t \leftarrow t + 1$
end while
Audit disjoint sentinels without updating D; gate $h^*(D)$
Emit profile, policy, evidence, manifest, and human-readable report
return preferred levels, NEUTRAL, or DEFER with confidence

Adaptive probe selection. We initialize with a budget-feasible, cost-adjusted D-optimal set, soft coverage rewards, and repeated anchors. For a feasible new or repeated block a, let $\Delta_t^R(a)$ denote its predicted reduction in Bayes regret $\mathcal{R}(D) = \mathbb{E}[R(h^*(D), \Theta_m) \mid D]$, $U_t(a)$ task-family uncertainty adjusted for repeats, $G_t(a)$ added coverage, and $\eta(a)$ semantic reliability. We select

$$a_t^* = \arg\max_{a \in \mathcal{A}_t} \frac{\eta(a)\,[\Delta_t^R(a) + \omega U_t(a) + \lambda G_t(a)]}{C(a)}, \tag{7}$$

Here $C(a)$ and ω, λ are fixed. Unlike EIG, Eq. (7) targets uncertainty that can change a representation decision. Components and hyperparameters are in Appendix C.3.

Stopping, robustness, and artifacts. Here $\Gamma(D)$ is minimum normalized coverage across tracks, factors, cores, and families. Once $\Gamma(D) = 1$, diagnosis stops after two posterior states with regret below $\epsilon_R = 0.02$, when the best score falls below $\epsilon_A = 0.001$, or upon candidate or budget exhaustion. Attempted blocks are charged; only complete blocks update the posterior. Disjoint sentinels are audited after stopping. Profile and policy files drive downstream use; retry rules and schemas are in Appendix C.3–C.4.

Profile-conditioned adaptation. For an existing Skill, a controller compares its representation fingerprint with the source-independent policy: matches or neutral evidence yield KEEP, confident mismatches TRANSFORM, and weak, uncovered, or heterogeneous evidence DEFER. The resulting `transformation_spec.json` constrains edits or a fixed stronger generator. Candidates must preserve invariants and pass structural, semantic, task, and non-inferiority checks; failed repairs roll back to the immutable source. This interface supports migration, cold start, generation, and drift repair. Schemas and validation gates are in Appendix C.4.

6 Experiments and Analysis

Research questions. We ask whether representation preferences are model-specific and reproducible (**RQ1**) and recoverable under limited budget (**RQ2**). We then test their value for migration (**RQ3**) and broader applications (**RQ4**), and identify the components responsible (**RQ5**).

Models, budgets, and evaluation. We profile seven deployments and use Qwen3.5-27B→9B as the primary migration setting; seven additional transfers appear in Table 3. Diagnosis uses 10/20/40% of expected full-bank cost (20% primary) across three seeds, with 100% as reference. Appendix D.2 specifies the models, metrics, cost accounting, and deployment manifests.

Benchmarks and baselines. We evaluate migration on ALFWorld, WebShop, SkillsBench, and SWE-Skills-Bench, plus search-augmented QA. Matched comparisons share budgets, source skills, generators, validators, and blind tests. Appendix D.2 details datasets, baselines, and native configurations.

Method	ALFWorld							WebShop		SkillsBench	SWE-Skills	
	Pick	Look	Clean	Heat	Cool	Pick2	All	Score	Succ.	Pass	Pass	Avg.Rank↓
Qwen3.5-9B (No Skill)	42.3	31.7	28.3	20.2	16.8	13.1	25.9	33.0	12.5	36.4	28.6	19.25
ReAct	53.2	39.6	38.2	24.8	26.4	23.7	35.9	48.7	21.5	39.0	30.1	17.25
GRPO	91.6	68.0	90.5	76.0	74.4	66.9	79.3	80.6	67.8	44.1	34.9	11.00
SimpleMem+GRPO	90.0	64.7	62.5	53.6	66.6	29.7	64.1	69.1	49.0	45.8	35.3	12.25
SkillRL	97.0	72.8	91.4	90.6	<u>95.2</u>	<u>87.6</u>	90.2	86.0	73.8	52.4	40.4	7.75
SkillGraph	**99.8**	<u>80.0</u>	<u>98.1</u>	<u>98.7</u>	82.7	85.4	<u>91.1</u>	**92.1**	**84.7**	54.9	41.1	5.25
Original Skill	46.2	34.8	30.1	18.0	14.6	12.3	25.8	34.5	12.8	35.7	27.4	19.75
GEPA	88.7	67.2	84.7	72.0	70.9	61.3	74.2	78.5	64.4	48.7	36.8	10.50
SkillReducer	84.3	64.2	80.3	68.2	66.1	58.4	70.3	75.2	60.3	49.3	38.9	10.75
SKILLPROBE Migration	96.3	76.3	94.9	92.6	93.8	86.9	90.0	88.4	78.4	58.8	45.5	5.25
GEPA+**SKILLPROBE**	97.4	77.4	95.8	93.6	94.7	86.7	90.0	89.6	79.9	59.4	45.1	5.00
SkillReducer+**SKILLPROBE**	96.5	75.7	94.4	92.4	92.4	85.3	89.1	88.6	77.8	60.7	<u>46.7</u>	5.25
SkillRL+**SKILLPROBE**	98.1	78.3	95.7	95.5	**96.0**	**88.5**	91.0	90.6	82.7	<u>61.1</u>	**47.5**	<u>2.75</u>
SkillGraph+**SKILLPROBE**	<u>99.1</u>	**81.5**	**99.4**	**99.8**	83.6	84.1	**91.5**	<u>92.0</u>	<u>84.5</u>	**62.4**	**47.5**	**1.50**

Table 2: Primary migration Qwen3.5-27B→Qwen3.5-9B. ALFWorld reports six task types and aggregate success; WebShop reports score/success; Skills and SWE report pass rate. Average rank uses ALF All, Web Succ., Skills, and SWE. Bold/underline are data-derived best/second-best among shown comparable methods.

Source → Target	ALF All	Web Succ.	Skills	SWE
Within-family transfer				
Gemma 2 27B → 9B	90.6/92.1	82.6/83.2	53.0/58.6	39.2/44.6
Llama 3.1 70B → 8B	88.2/91.3	86.1/85.4	56.7/60.8	42.3/46.5
Cross-family transfer				
Qwen3.5-27B → Gemma 2 9B	72.8/87.5	68.9/75.1	44.3/52.7	33.0/42.1
Gemma 2 27B → Qwen3.5-9B	76.4/90.7	71.3/73.5	45.9/51.6	38.4/44.6
Qwen3.5-27B → Llama 3.1 8B	69.7/85.2	65.6/76.8	48.8/56.2	35.5/39.3
Llama 3.1 70B → Qwen3.5-9B	73.1/88.4	79.2/80.4	46.5/55.4	39.7/45.8
Open-to-closed transfer				
Qwen3.5-27B → GPT-4o	91.4/93.2	88.1/89.3	58.1/65.2	45.8/52.5

Table 3: Additional within-family, cross-family, and open-to-closed migration results. Each metric cell reports SkillGraph / SkillGraph+**SKILLPROBE** on the same target deployment.

6.1 REPRESENTATION PREFERENCES DIFFER

Figure 1 answers RQ1: equivalent representations produce distinct response patterns. Qwen3.5-2B has negative effects for granularity (T3), examples (T8), and resource organization (E6), whereas 27B is positive on all three. Moreover, 35B-A3B is closer to 9B than 27B (0.076 versus 0.247), so preference is not monotonic in parameter count. The leave-one-core-out classifier identifies seven deployments with 75.7% macro accuracy; most confusion occurs between Qwen variants. Because it sees only held-out preference effects, this result establishes reproducible representation fingerprints, not general model identification.

6.2 LOW-BUDGET DIAGNOSIS

Figure 3 approaches the full-bank reference smoothly. At 20% (Table 4(a)), **SKILLPROBE** reduces Continuous CAT's error from 30.5 to 11.9, raises agreement by 17.0 points and sentinel accuracy by 7.3, and uses 18.9% fewer tokens. It saves 84.5% tokens relative to Full Bank, with a 4.5-point sentinel gap.

Budget behavior. At 10%, **SKILLPROBE** reaches 69.4% agreement and 85.9% sentinel accuracy, versus 52.7% and 75.6% for continuous CAT. At 40%, agreement reaches 95.1% and sentinel accuracy 95.8%, near the full-bank reference. The primary run consumes 198k tokens, 15.5% of Full Bank. D-optimal ignores terminal decisions, while EIG can prioritize action-irrelevant uncertainty; continuous CAT lacks paired blocks, interactions, and worst-family regret. Their ordering supports the decision-aware design. Nevertheless, sentinel accuracy remains below Full Bank, so **SKILL-**

(a) Diagnosis

Method	Bud.	Err.↓	Agr.↑	Sent.↑	Tok.↓
Full Bank	100	6.1	98.8	97.4	1280
Fixed D-opt.	20	30.5	53.8	75.2	247
Bayesian EIG	20	39.8	49.3	81.6	266
IRT/OED	20	39.3	49.6	84.2	221
Continuous CAT	20	30.5	67.3	85.6	244
SKILLPROBE	20	**11.9**	**84.3**	**92.9**	**198**

(b) Ablation at 20% budget

Setting	Err.↓	Agr.↑	Migr.↑	Tok.↓
SKILLPROBE (full)	11.9	84.3	92.4	198
w/o Profile	15.4	87.8	88.6	191
w/o Regret	12.7	84.1	83.2	196
w/o Interactions	16.8	86.9	86.0	171
w/o Stopping	11.7	91.8	92.0	284

Table 4: Low-budget diagnosis and component ablation. Panel (a) compares selected 20% methods with the 100% Full Bank. IRT/OED is the cost-aware IRT/OED baseline. Panel (b) shows representative 20% ablations. Tokens are in thousands. Complete results are in Appendix Tables 11 and 17.

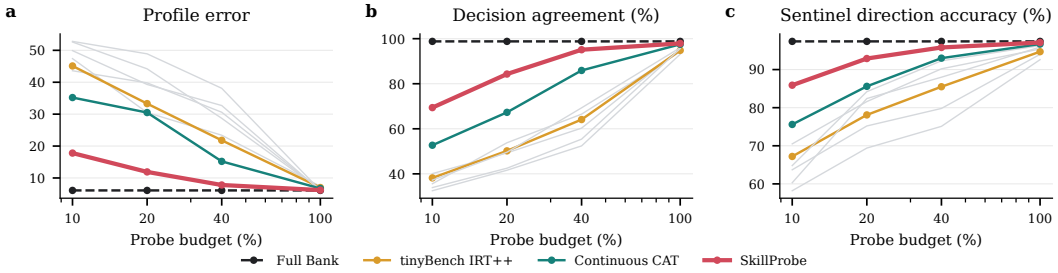


Figure 3: Diagnosis quality versus probe budget. **SKILLPROBE** is highlighted. Error decreases while decision agreement and sentinel direction accuracy increase.

PROBE confidence-gates the resulting profile. This answers RQ2 without claiming equivalence to exhaustive probing.

6.3 CORE APPLICATION: SKILL MIGRATION

Table 2 answers RQ3. The original 27B skill slightly harms 9B on ALFWorld (25.8 versus 25.9 without skill) and SWE (27.4 versus 28.6). In contrast, **SKILLPROBE** migration reaches 90.0 ALFWorld, 78.4 WebShop success, 58.8 SkillsBench, and 45.5 SWE. Conditioning GEPA, SkillReducer, SkillRL, and SkillGraph improves SkillsBench by 7.5–11.4 points and SWE by 6.4–8.3. SkillGraph+**SKILLPROBE** ranks first (1.50), followed by SkillRL+**SKILLPROBE** (2.75). Gains are largest on procedural benchmarks and smaller for saturated interactive policies. Indeed, the best combination trails SkillGraph by 0.2 WebShop-success points and 0.1 score points, bounding the claim of universal improvement.

Table 3 extends RQ3 beyond Qwen scaling. SkillGraph+**SKILLPROBE** scores higher than SkillGraph in 27 of 28 paired benchmark comparisons; the sole decline is 0.7 points in WebShop success for Llama 3.1 70B→8B. All 16 comparisons across reciprocal Qwen–Gemma and Qwen–Llama transfers improve by 1.2–15.5 points, and all four Qwen3.5-27B→GPT-4o comparisons improve by 1.2–7.1 points. These results support migration across the tested open families and to one closed-model target.

6.4 SEARCH QA GENERALIZATION

Only NQ and HotpotQA provide construction examples, so the remaining five datasets test transfer. In Table 5, SkillGraph+**SKILLPROBE** leads macro EM at 53.0 (+2.3); SkillRL+**SKILLPROBE** gains 2.6 and EvolveR+**SKILLPROBE** gains 6.5. Improvements span all out-of-domain sets, although ZeroSearch and StepSearch remain best on PopQA and MuSiQue. The profile therefore complements, rather than replaces, specialized retrieval and search policies.

6.5 BROADER APPLICATIONS AND ABLATIONS

Figure 4 answers RQ4 across independent downstream procedures. At 20%, cold start on held-out 4B reaches 52.8 SkillsBench and closes 81.8% of the no-profile/full-bank gap. Guided generation

Method	Single-Hop			Multi-Hop				Avg.
	NQ†	TQA∗	PQA∗	HQA†	2W∗	MuS∗	Bam∗	
Qwen3.5 Base	14.6	38.2	3.5	19.7	25.8	6.3	17.6	18.0
RAG	30.8	60.9	20.7	28.4	25.6	11.2	19.0	28.1
Search-R1	42.7	63.0	42.3	39.2	42.9	16.7	39.9	41.0
ZeroSearch	46.4	63.8	**54.2**	37.7	37.0	20.4	30.6	41.4
StepSearch	40.5	60.8	38.7	41.8	39.9	**25.2**	43.6	41.5
EvolveR	46.4	65.8	47.2	40.9	44.3	17.3	57.4	45.6
SkillRL	48.1	65.0	48.6	45.6	42.7	22.3	75.8	49.7
SkillGraph	50.1	65.5	50.7	46.4	46.7	21.2	74.4	50.7
IGRPO	47.5	64.1	49.1	44.3	45.4	23.5	68.4	48.9
SKILLPROBE Migration	50.4	65.0	51.4	46.9	45.6	24.3	76.8	51.5
EvolveR+**SKILLPROBE**	51.8	66.0	52.2	47.7	46.0	24.9	76.1	52.1
SkillRL+**SKILLPROBE**	51.4	66.6	52.3	47.5	46.3	24.0	**77.9**	52.3
SkillGraph+**SKILLPROBE**	**52.7**	**66.8**	53.8	**48.4**	**47.3**	24.6	77.1	**53.0**

Table 5: Exact match on search-augmented QA. †/∗ mark in-/out-of-domain evaluation.

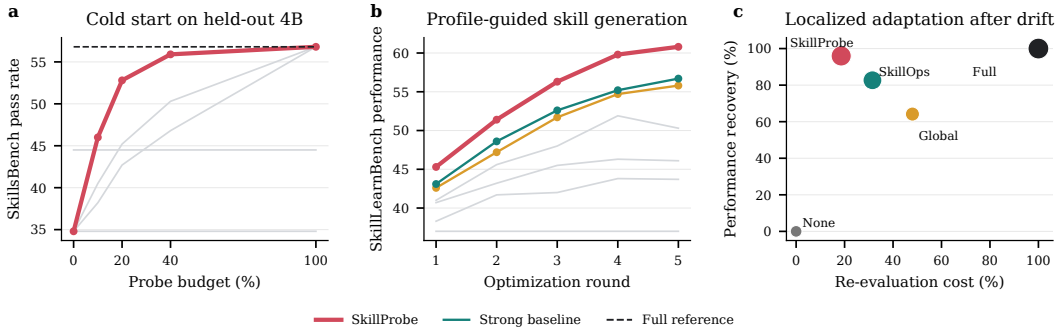


Figure 4: Three independent uses of a diagnosed profile. (a) Cold start on held-out Qwen3.5-4B. (b) iterative generation on SkillLearnBench. (c) version-drift adaptation: lower reevaluation cost and higher recovery are better; marker area encodes affected-skill localization.

on SkillLearnBench reaches 56.3 by round 3, nearly SkillGen's round-5 score (56.7), and finishes at 60.8. For Qwen3.5→3.6 drift, local repair recovers 96.0% performance at 18.6% reevaluation cost with 92.4% localization, versus 64.1% at 47.2% for global adaptation. Marker area denotes localization, not uncertainty.

Panel (b) of Table 4 summarizes RQ5; the full ablation results are in Appendix Table 17. Removing decision regret causes the largest migration drop (9.2); removing interactions loses 6.4 and raises error to 16.8. Removing cost normalization or stopping raises tokens by 32.3% or 43.4% without improving migration. Notably, removing Bayesian profiling increases agreement to 87.8% but worsens both profile error and migration, confirming that agreement alone is insufficient. Thus profiling, regret, and interactions drive quality, while normalization and stopping primarily improve efficiency.

7 CONCLUSION

We introduced **SKILLPROBE**, a budget-aware framework for diagnosing model-specific preferences over semantically equivalent skill representations. Its dual-track probe bank and adaptive Bayesian selection produce actionable profiles without exhaustive evaluation. These profiles improve strong skill-migration baselines by up to 11.4 points on SkillsBench and 8.3 points on SWE-Skills-Bench, while complementary experiments demonstrate broader applicability. Overall, our results show that diagnosing representation preferences can make skill adaptation more effective and economical.

Our current evaluation remains limited to text-based skills and a fixed set of agent scaffolds. Future work will investigate online profile updates under continual model and scaffold drift and extend **SKILLPROBE** to multimodal and longer-horizon tool environments.

AI USE STATEMENT

We used generative AI tools to assist with editing and polishing the manuscript's language and with LaTeX formatting. The authors reviewed the AI-assisted changes and take full responsibility for the final content, including its claims and results.

REFERENCES

Lakshya A. Agrawal, Shangyin Tan, Dilara Soylu, Noah Ziems, Rishi Khare, Krista Opsahl-Ong, Arnav Singhvi, Herumb Shandilya, Michael J. Ryan, Meng Jiang, et al. Gepa: Reflective prompt evolution can outperform reinforcement learning. *arXiv preprint arXiv:2507.19457*, 2025.

Esma Balkir, Alice Pernthaller, Marco Basaldella, José Hernández-Orallo, and Nigel Collier. Confident rankings with fewer items: Adaptive llm evaluation with continuous scores. *arXiv preprint arXiv:2601.13885*, 2026.

Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, 10(3):273–304, 1995.

Le Chen, Erhu Feng, Yubin Xia, and Haibo Chen. Skillrt: Compiling skills for efficient execution everywhere. *arXiv preprint arXiv:2604.03088*, 2026.

Yudong Gao, Zongjie Li, Yuanyuanyuan, Zimo Ji, Pingchuan Ma, and Shuai Wang. Skillreducer: Optimizing llm agent skills for token efficiency. *arXiv preprint arXiv:2603.29919*, 2026.

Tingxu Han, Yi Zhang, Wei Song, Chunrong Fang, Zhenyu Chen, Youcheng Sun, and Lijie Hu. Swe-skills-bench: Do agent skills actually help in real-world software engineering? *arXiv preprint arXiv:2603.15401*, 2026.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6609–6625, 2020.

Valentin Hofmann, David Heineman, Ian Magnusson, Kyle Lo, Jesse Dodge, Maarten Sap, Pang Wei Koh, Chun Wang, Hannaneh Hajishirzi, and Noah A. Smith. Fluid language model benchmarking. *arXiv preprint arXiv:2509.11106*, 2025.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pp. 1601–1611, 2017.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, et al. Dspy: Compiling declarative language model calls into state-of-the-art pipelines. In *International Conference on Learning Representations*, 2024.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Xiangyi Li, Wenbo Chen, Yimin Liu, Shenghan Zheng, Xiaokun Chen, Yifeng He, Yubo Li, Bingran You, Haotian Shen, Jiankai Sun, et al. Skillsbench: Benchmarking how well agent skills work across diverse tasks. *arXiv preprint arXiv:2602.12670*, 2026a.

Xiaoyuan Li, Moxin Li, Keqin Bao, Yubo Ma, Wenjie Wang, Dayiheng Liu, and Fuli Feng. Skillgraph: Skill-augmented reinforcement learning for agents via evolving skill graphs. *arXiv preprint arXiv:2605.12039*, 2026b.

Yujian Liu, Jiabao Ji, Li An, Tommi Jaakkola, Yang Zhang, and Shiyu Chang. How well do agentic skills work in the wild: Benchmarking llm skill usage in realistic settings. *arXiv preprint arXiv:2604.04323*, 2026.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pp. 8086–8098, 2022.

Yuchen Ma, Yue Huang, Han Bao, Haomin Zhuang, Swadheen Shukla, Michel Galley, Xiangliang Zhang, and Stefan Feuerriegel. Skillgen: Verified inference-time agent skill synthesis. *arXiv preprint arXiv:2605.10999*, 2026.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pp. 9802–9822, 2023.

Krista Opsahl-Ong, Michael J. Ryan, Josh Purtell, David Broman, Christopher Potts, Matei Zaharia, and Omar Khattab. Optimizing instructions and demonstrations for multi-stage language model programs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2024.

Yipeng Ouyang, Yi Xiao, Yuhao Gu, and Xianwei Zhang. Skcc: Portable and secure skill compilation for cross-framework llm agents. *arXiv preprint arXiv:2605.03353*, 2026.

Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinybenchmarks: Evaluating llms with fewer examples. In *International Conference on Machine Learning*, 2024.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 5687–5711, 2023.

Hongji Pu, Xinyuan Song, and Liang Zhao. Skillops: Managing llm agent skill libraries as self-maintaining software ecosystems. *arXiv preprint arXiv:2605.13716*, 2026.

Qwen Team. Qwen3.5 model collection. Hugging Face model cards, 2026a. URL `https://huggingface.co/collections/Qwen/qwen35`.

Qwen Team. Qwen3.6-27b model card. Hugging Face, 2026b. URL `https://huggingface.co/Qwen/Qwen3.6-27B`.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design. In *International Conference on Learning Representations*, 2024.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Maksim Shaposhnikov, Nicolas Fortuin, Simon Stipcich, Maria I. Gorinova, Amy Heineike, and Rob Willoughby. A framework for evaluating agentic skills at scale. *arXiv preprint arXiv:2606.17819*, 2026.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. Alfworld: Aligning text and embodied environments for interactive learning. In *International Conference on Learning Representations*, 2021.

Hao Sun, Zile Qiao, Jiayan Guo, Xuanbo Fan, Yingyan Hou, Yong Jiang, Pengjun Xie, Fei Huang, and Yan Zhang. Zerosearch: Incentivize the search capability of llms without searching. *arXiv preprint arXiv:2505.04588*, 2025.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Musique: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=ehfRiF0R3a.

Rong Wu, Xiaoman Wang, Jianbiao Mei, Pinlong Cai, Daocheng Fu, Cheng Yang, Licheng Wen, Xuemeng Yang, Yufan Shen, Yuxin Wang, and Botian Shi. Evolver: Self-evolving llm agents through an experience-driven lifecycle. *arXiv preprint arXiv:2510.16079*, 2025.

Peng Xia, Jianwen Chen, Hanyang Wang, Jiaqi Liu, Kaide Zeng, Yu Wang, Siwei Han, Yiyang Zhou, Xujiang Zhao, Haifeng Chen, et al. Skillrl: Evolving agents via recursive skill-augmented reinforcement learning. *arXiv preprint arXiv:2602.08234*, 2026.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Bb4VGOWELI.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380, 2018.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. In *Advances in Neural Information Processing Systems*, 2022.

Hanrong Zhang, Shicheng Fan, Henry Peng Zou, Yankai Chen, Zhenting Wang, Jiayu Zhou, Chengze Li, Wei-Chieh Huang, Yifei Yao, Kening Zheng, et al. Evoskills: Self-evolving agent skills via co-evolutionary verification. *arXiv preprint arXiv:2604.01687*, 2026.

Xuhui Zheng, Kang An, Ziliang Wang, Yuhang Wang, and Yichao Wu. Stepsearch: Igniting llms search ability via step-wise proximal policy optimization. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 21805–21830, 2025.

Shanshan Zhong, Yi Lu, Jingjie Ning, Yibing Wan, Lihan Feng, Yuyi Ao, Leonardo F. R. Ribeiro, Markus Dreyer, Sean Ammirati, and Chenyan Xiong. Skilllearnbench: Benchmarking continual learning methods for agent skill generation on real-world tasks. *arXiv preprint arXiv:2604.20087*, 2026.

A TERMINOLOGY AND NOTATION

The main paper defines the four concepts needed to understand the central claim. A *skill representation preference* is a deployment-conditioned ordering of semantically equivalent realizations by observed performance. A *semantic core* is the invariant goal, operations, branches, constraints, validation, and recovery behavior shared by those realizations. *Operational equivalence*, written $r_i \sim_z r_j$, means that two representations preserve the same core and differ only on declared factors; it is established by the construction protocol, not by a formal equivalence proof. A *deployment profile* is the posterior over these factor effects for one fully specified model–agent deployment. It is not a subjective preference, aggregate capability score, or permanent property of a base model.

Table 6 defines auxiliary terms used for implementation and reporting. These terms refine the four central concepts but do not introduce additional claims.

A.1 OBSERVATION AND UTILITY SCHEMA

Table 7 expands the run observation Y used by the main paper. All fields are stored before paired effects are computed. Score ranges and transformations are frozen with the benchmark; downstream results do not tune them.

For example, suppose a request completes but returns invalid JSON after 600 tokens and 1,200 ms. Its record is $Y = (0, 0, 0, 600, 1200, 1, 0, \texttt{invalid_json})$, yielding utilities $(0, 0, 0, -\log 601, -\log 1201, 0)$. The low token count cannot compensate for invalid behavior because channels are modeled separately and task quality is the primary action channel. Conversely, a feasible but wrong answer retains $A_{\text{feas}} = 1$ and an explicit wrong-answer type while receiving its verifier-determined task score. This distinction separates inability to satisfy the execution contract from ordinary task error.

A.2 SYMBOL INDEX

Table 8 collects notation that is distributed across the problem formulation and method equations. A subscript m always denotes one complete deployment identity, not only a model checkpoint. Indices q, k, and g denote a representation contrast, utility channel, and task family, respectively; o indexes one complete fixture–repeat comparison.

B SKILLPROBEBANK CONSTRUCTION

B.1 SCOPE AND COMPOSITION

SKILLPROBEBANK is a static, manually designed benchmark for controlled interventions on skill representation. It contains 54 semantic cores: 36 selectable probe-pool cores and 18 disjoint sentinel cores. The bank materializes 270 deterministic task fixtures (54 smoke and 216 scored), 156 candidate probes, and 360 model-facing variants. The instruction track contains 24 pool cores, 96 main-effect probes, 18 interaction probes, and 264 variants. The executable track contains 12 pool cores, 36 main-effect probes, six interaction probes, and 96 variants. Sentinels are excluded from adaptive selection and profile fitting; they only test out-of-core direction accuracy.

Synthetic deterministic tasks were selected to isolate representation effects from changing world knowledge, retrieval, or environment state. Downstream claims use separately maintained public benchmarks and never use **SKILLPROBEBANK** task fixtures as feedback. The bank contains no personal, sensitive, or restricted third-party data.

The taxonomy was selected by decomposing a Skill package into two actionable classes of decisions: textual serialization of fixed information units and executable placement or enforcement of fixed operations, interfaces, checks, recovery, and resources. A candidate factor was retained only if its levels could be manipulated independently without changing the goal, I/O contract, available tools, permissions, or failure semantics. The factors were frozen before target-model evaluation and are intentionally not an exhaustive account of Skill design.

Term	Operational meaning	Role or boundary
Representation factor / level	A predeclared design dimension and one of its admissible settings, such as hierarchy depth or validation placement.	Only the declared factor may change within a contrast.
Matched variant	One textual, executable, or mixed realization of a semantic core at specified factor levels.	It is a skill representation, not a separate task.
Matched probe block	The variants, shared task fixtures, and blind verifier required to estimate one main or interaction contrast.	The unit selected by the adaptive policy.
Complete block	A block for which every required contrast cell was executed under the same deployment and seed.	Incomplete blocks never update the posterior.
Blind verifier	A scorer that receives outputs and task evidence without factor labels, variant identity, or expected direction.	Prevents the scorer from rewarding the intended hypothesis.
Main / interaction effect	A paired two-cell difference or a four-cell difference-in-differences, respectively.	Defined formally in the effect-extraction equations of the main paper.
Typed failure	A discrete outcome category such as invalid contract, tool error, timeout, or unrecovered execution failure.	Retained separately from continuous quality scores.
Feasibility gate	A deterministic rule that marks critical contract or execution violations infeasible before metric comparison.	Prevents fluent text from compensating for invalid behavior.
Probe pool	Cores and blocks eligible for initialization, adaptive selection, and posterior fitting.	Selection may use their outcomes.
Sentinel set	Disjoint held-out cores excluded from selection and fitting.	Used only for generalization and drift checks.
Anchor / interleaved repeat	An anchor is a small block intentionally repeated to estimate noise; an interleaved repeat revisits a block after intervening actions.	Anchors use three executions in total; all other repeats share the same three-execution cap.
Coverage	Minimum normalized completion across tracks, factors, semantic cores, and task families, as defined in the main paper.	A hard early-stopping gate, not a performance metric.
Semantic-risk score	A preregistered audit annotation for the chance that a proposed factor change leaks into core semantics.	Discounts acquisition reliability and is frozen before model evaluation.
Family deviation / heterogeneity	A task-family-specific departure from the global factor effect; heterogeneity denotes credible disagreement across families.	Blocks an unjustified global preference.
ROPE / stability	The region of practical equivalence defines negligible effects; stability summarizes posterior direction and repeat consistency.	Used for confidence-gated actions, not significance testing.
Decision regret	Loss relative to the representation state an oracle with the true deployment profile would recommend.	The acquisition target in the main objective.
KEEP / TRANSFORM / DEFER	Retain the current representation, apply an approved representation change, or request more evidence / human handling.	Derived after comparing a source-independent recommendation with a Skill fingerprint.

Table 6: Auxiliary terminology used by **SKILLPROBEBANK** and **SKILLPROBE**.

Field	Stored value and domain	Utility map or treatment
S_{task}	Blind-verifier task score in $[0, 1]$; exact match, required-element recall, or set F1 according to the preregistered task verifier.	$u_{\mathrm{task}}(Y) = S_{\mathrm{task}}$; primary channel for representation actions.
S_{contract}	Output-contract adherence in $[0, 1]$.	$u_{\mathrm{contract}}(Y) = S_{\mathrm{contract}}$.
S_{exec}	Execution validity in $[0, 1]$, including executable assertions where present.	$u_{\mathrm{exec}}(Y) = S_{\mathrm{exec}}$.
C_{tok}	Nonnegative sum of input and output tokens for the request.	$u_{\mathrm{tok}}(Y) = -\log(1 + C_{\mathrm{tok}})$; raw tokens also enter $C(a)$ and budget accounting.
C_{lat}	Nonnegative wall-clock request latency in milliseconds.	$u_{\mathrm{lat}}(Y) = -\log(1 + C_{\mathrm{lat}})$; raw latency remains in the run record.
A_{req}	Binary indicator that the provider returned a model response.	Combined with verifier feasibility in the reliability channel.
A_{feas}	Binary verifier gate for a parseable output satisfying critical contract and execution requirements.	$u_{\mathrm{rel}}(Y) = A_{\mathrm{req}} A_{\mathrm{feas}}$. If zero, verifier quality scores are zeroed before effect extraction.
F	Nullable typed category, e.g., invalid JSON, contract violation, wrong answer, timeout, tool error, or unrecovered provider failure.	Retained categorically in raw evidence and failure summaries; never encoded as an arbitrary scalar severity.

Table 7: Complete per-run observation schema and the six utility channels. No row is combined with another through a learned or hand-tuned weight.

Instruction cores cover six families: structured extraction, constraint-preserving transformation, rule classification, multi-stage planning, verification/audit, and file/tool operation. The probe pool contains four cores per family and the sentinel set contains two. Executable cores cover structured file transformation, table aggregation, batch file operation, and configuration validation; the pool contains three cores per family and every family is represented in the six executable sentinels. The split is performed at the semantic-core level, so tasks and any derived artifacts cannot cross splits.

Symbol	Meaning	Scope or convention
$z, r, r_i \sim_z r_j$	Semantic core, its realization, and operational equivalence	Equivalence is established by the frozen construction protocol.
m, Φ_m, Θ_m	Deployment identity, posterior profile, and latent factor effects	A scaffold, prompt, tool, permission, or revision change creates a new m.
$\mathcal{Q}, \mathcal{K}, \mathcal{G}$	Contrasts, utility channels, and task families	$\mathcal{Q}$ includes 14 main factors and four confirmatory interactions.
$Y, u_k(Y), \Delta_{o,k}$	Run observation, channel utility, and paired effect	All utilities are larger-is-better; channels remain separate.
$\mu_{q,k}, b_{q,k,g}$	Global contrast effect and family deviation	$\theta_{q,k,g} = \mu_{q,k} + b_{q,k,g}$.
$\sigma^2_{g,k}, \tau^2_{q,k}$	Observation noise and between-family variance	Both are inferred independently for every utility channel.
$D_t, a_t, n_t(a)$	Complete effects, candidate block, and attempted executions	Only complete blocks enter D_t; every attempt is charged.
$C(a), B, b_t$	Frozen relative action cost, total budget, and remaining budget	Realized tokens and latency are reported separately.
h_q, ρ_q, e_q	Representation action, ROPE, and effective effect	$h_q \in \{-1, 0, +1\}$; DEFER is a later abstention.
$\mathcal{R}(D), \Delta^R_t(a)$	Posterior Bayes regret and its expected reduction	The acquisition target is terminal decision quality.
$U_t(a), G_t(a), \eta(a)$	Robust uncertainty, coverage gain, and semantic reliability	Auxiliary acquisition terms; none is a task reward.
$\Gamma(D)$	Minimum normalized coverage	A hard gate for confidence-based stopping.

Table 8: Notation used in the main paper and appendix.

ID	Factor	Levels	Controlled invariant
T1	Organization format	prose; numbered; fielded	wording and information units
T2	Hierarchy depth	flat; two-level; three-level	section content and order
T3	Procedural granularity	merged; standard; atomic	atomic operations
T4	Control-flow encoding	narrative; if–then; pseudocode	conditions, branches, termination
T5	Contract encoding	prose; dedicated; typed schema	fields, types, failure conditions
T6	Constraint salience	central; near-use; both	constraint content
T7	Validation placement	terminal; step-local; separate	predicates and failure responses
T8	Example organization	concentrated; adjacent; appendix	examples and labels
E1	Code externalization	inline; external	executable statements and dependencies
E2	Module decomposition	monolith; modules	call graph and operations
E3	Action granularity	high-level; atomic tools	available operations and behavior
E4	Validation enforcement	agent; assertions; dual	validation predicates
E5	Recovery placement	text; code; layered	triggers, recovery, fallback
E6	Resource organization	embedded; external	resource bytes and lookup keys

Table 9: Predeclared representation factors. Levels change serialization, placement, or enforcement while preserving the semantic core.

Preregistered confirmatory interactions are T1×T5 (organization/contract), T2×T3 (hierarchy/granularity), T4×T7 (flow/validation), and E1×E2 (code location/modules). All others are exploratory and strongly shrunk. Only the four preregistered pairs receive dedicated four-cell confirmatory probes.

The selectable inventory is a balanced incomplete-block design at the semantic-core level. Each instruction main factor appears in 12 pair probes and each executable main factor in six. Each confirmatory interaction appears in six four-cell probes. A core instantiates only a subset of factors, preventing one task template from dominating all effects, while every factor is distributed over multiple cores and families. The design is balanced over declared factors, not over every possible factor-level pair; the exact cells remain explicit in the released design matrix.

B.2 Semantic Core and Probe Records

A semantic core declares a goal, preconditions, ordered operations, branches, constraints, output contract, validation predicates, recovery behavior, and allowed resources. A variant references one core and changes only declared factor levels. A main-effect probe binds two variants; an interaction probe binds the four cells of a 2×2 factorial contrast. Each probe also stores shared task IDs, a blind verifier ID, relative cost, semantic-risk annotation, track, factor IDs, and stable immutable identifiers. A model-facing artifact is Markdown stored in JSON Lines; executable artifacts may additionally reference scripts and resources. Thus a probe is neither merely text nor merely code: it is a typed multi-level object whose rendered payload may be textual, executable, or mixed.

The record hierarchy is *core* → *variant* → *probe* → *task* → *run*. Core YAML is normative. Variant records point to rendered artifacts and enumerate all semantic-unit IDs that must be present. Probe records bind cells to shared scored tasks and a verifier. Task records contain synthetic inputs and ex-

pected structures, whereas run records contain deployment identity, randomized order, repeat index, raw output, multichannel outcome, validity, and cost. This separation allows the static Bank to be versioned without embedding observations from any evaluated model.

For a concrete main-effect example, core `SPB-PP-I-EXT-01` specifies contact-record extraction with output fields `full_name`, `email`, and `team`. Probe `SPB-Q-I-ME-001` renders the same ten semantic units as continuous prose and numbered steps, changes only T1, and evaluates both cells on the same four exact-JSON fixtures. Its `estimated_cost` stores static relative token and execution priors frozen before observation; measured token use and latency remain run outcomes.

The release uses YAML for definitions, JSON Lines for records, CSV for the design matrix, JSON Schema Draft 2020-12 for validation, and SHA-256 manifests. Stable IDs are immutable, and construction data are versioned separately from observations.

B.3 Semantic Equivalence and Admission

Construction follows the protocol below; it is a data-generation protocol, not part of adaptive diagnosis or downstream skill adaptation.

1. Freeze the semantic core and enumerate information units before rendering any variant.
2. Apply deterministic factor-specific renderers. No renderer may add knowledge, tools, permissions, operations, checks, examples, or failure behavior.
3. Validate schema, references, required units, and task fixtures automatically.
4. Audit each contrast for goal, contract, operation, constraint, validation, recovery, and resource equivalence. Length is matched within 5% where feasible; otherwise it is retained as a mediator.
5. Run smoke fixtures, then scored deterministic fixtures with a verifier that is blind to variant identity.
6. Admit a probe only if all cells are valid, non-inferior under deterministic checks, and free of factor leakage. Failed items are repaired and re-audited; unresolved items are excluded.

Release eligibility requires two independent reviewers to complete the audit template. Disagreements are adjudicated against the semantic core by a third reviewer, not against model outcomes; raters do not see execution results. The final public archive must retain the two raw decisions and any adjudication row for every admitted variant. Automated release validation separately checks inventory, cell cardinality, schema conformance, and referential integrity and reports no structural errors for the frozen inventory. This automated result does not establish semantic equivalence by itself. Probe-pool and sentinel cores are disjoint, and sentinel results may not tune the bank, selector, profile, or adaptation policy.

Length matching is a control rather than an admission target that can override semantics. Rendered cells are matched within 5% when the declared factor permits it. Otherwise, exact token counts and the reason for the mismatch remain in the audit row and length is treated as a recorded mediator. No content is deleted solely to satisfy the length window, and a variant is excluded whenever matching would remove a required semantic unit.

B.4 Scoring and Provenance

All 270 **SkillProbeBank** fixtures use one of four deterministic verifier families: exact-JSON matching, required-element checking, issue-set comparison, or file-bundle comparison with declared normalization. Verifiers receive task outputs and expected structures but not factor labels, variant identities, or expected preference directions. A blind LLM judge is therefore unnecessary for Bank scoring; model-based judges, when required by an external downstream benchmark, are isolated from Bank admission and profile construction.

Every record retains immutable bank-version, core, probe, variant, task, and verifier identifiers. Human-authored definitions use YAML, flat records use JSON Lines, the design matrix uses CSV, schemas follow JSON Schema Draft 2020-12, and release manifests include SHA-256 hashes.

C Full Diagnostic Protocol

C.1 Deployment Identity and Randomization

A profile is indexed by model identifier and revision, agent scaffold, system prompt hash, inference configuration, tool protocol, permissions, environment hash, and access time for mutable APIs. Changing any element creates a new deployment identity. Within a complete probe block, cell order and task order are randomized by the run seed. Incomplete blocks never update the posterior. Infrastructure retries are logged separately from model failures; invalid model outputs remain outcomes rather than being silently retried.

The implementation hashes the complete deployment object before pairing any runs and rejects effects whose hashes differ. Within repeat r, variant–task pairs are shuffled by a deterministic pseudorandom generator seeded with the experiment seed plus $1{,}000{,}003r$. Resume keys include variant, task, seed, repeat, and deployment hash. An infrastructure exception permits at most two additional attempts; the default retry delay is zero and provider-level backoff is recorded by the provider wrapper. Model refusal, invalid syntax, and tool misuse are outcomes rather than infrastructure exceptions.

The primary budgets are 10%, 20%, and 40% of expected full-bank cost, with 20% predeclared as primary and 100% used as a reference. All aggregate values average seeds 11, 23, and 47. Model outputs are capped at 4,096 tokens. Open models use deterministic decoding; reasoning APIs use a fixed medium setting. The example configuration deliberately leaves mutable provider revisions and access dates null so an unfrozen live endpoint cannot be mistaken for a final experimental deployment.

C.2 Posterior Model and Decision Rule

One complete fixture–repeat comparison is one observation o. Its channel-k effect $\Delta_{o,k}$ is associated with declared contrast $q(o)$ and task family $g(o)$. The profile model in the main paper uses

$$\theta_{q,k,g} = \mu_{q,k} + b_{q,k,g},$$
$$\Delta_{o,k} \mid \theta, \sigma^2 \sim \mathcal{N}\Big(\theta_{q(o),k,g(o)}, \sigma^2_{g(o),k}\Big).$$

The family deviation is indexed by contrast; it is not a single family intercept shared by all factors. The frozen, conditionally conjugate priors are

$$\begin{aligned} \mu_{q,k} &\sim \mathcal{N}(0,1), \\ b_{q,k,g} \mid \tau^2_{q,k} &\sim \mathcal{N}(0, \tau^2_{q,k}/\kappa_f), \\ \tau^{-2}_{q,k} &\sim \mathrm{Gamma}(2, 0.05), \\ \sigma^{-2}_{g,k} &\sim \mathrm{Gamma}(2, 0.05). \end{aligned}$$

where $\kappa_f = 4$ and Gamma uses the shape–rate parameterization. Posterior moments and predictive draws are obtained by conditionally conjugate updates. All hyperparameters are frozen before target-model evaluation and are shared across compared selection methods.

The released estimator uses a Gibbs sampler with 4,000 retained draws per utility channel. Burn-in is $\max(100, \min(500, N_{\mathrm{draw}}/2))$, yielding 500 burn-in iterations in the reported configuration. Each channel receives a deterministic seed offset, unobserved declared contrasts remain in the profile under the prior, and 95% credible intervals are empirical posterior quantiles. The adaptive candidate scorer uses a cheaper 320-draw task-quality refit; this approximation ranks candidates only and does not replace the final 4,000-draw profile.

For the primary task-quality decision, covered families and contrasts receive uniform normalized weights, $\nu_g = 1/|\mathcal{G}_q(D)|$ and $w_q = 1/|\mathcal{Q}|$; hence reported regret is on a normalized per-contrast scale. The task-quality ROPE is $\rho_q = 0.03$ for every predeclared q. The remaining profile channels use ROPEs 0.02 for contract, execution, and reliability, $\log(1.05)$ for token efficiency, and $\log(1.10)$ for latency efficiency. A preferred level requires posterior probability at least 0.95 beyond either side of its ROPE, evidence from at least three semantic cores and two task families, and no heterogeneous

family effect. Neutrality requires posterior probability at least 0.80 inside the ROPE. Confident opposing family directions at probability 0.80 mark the contrast heterogeneous.

The optimized action space remains $\{-1, 0, +1\}^{|\mathcal{Q}|}$. Outside the ROPE, regret is 0 for the oracle direction, $|e_q|$ for neutrality, and $2|e_q|$ for the opposite direction; all three actions tie when $e_q = 0$, with ties resolved to neutrality. DEFER is applied only after this Bayes decision when a confidence, evidence, coverage, or heterogeneity gate fails. It is therefore an abstention status rather than a fourth action hidden from the regret equation.

The readout separates a posterior state from an operational action. A contrast is `preferred` or `avoid` only when probability beyond the relevant ROPE is at least 0.95; it is `neutral` when probability inside the ROPE is at least 0.80; otherwise it is `unresolved`. Evidence from fewer than three cores or two families, failed global coverage, or opposing credible family directions converts any provisional recommendation to DEFER. These gates are applied after the Bayes action and therefore do not alter the regret objective.

C.3 INITIALIZATION, ACQUISITION, AND STOPPING

For factor-and-interaction incidence vector v_p, the initialization information matrix is

$$M(S) = 10^{-3} I + \sum_{p \in S} \frac{v_p v_p^\top}{C(p)}.$$

Initialization starts from a 14-block candidate design and greedily increases $\log\det M(S)$ with soft factor coverage weighted 2.0, track coverage 3.0, and semantic-core diversity 0.15. These terms promote breadth but do not require the initialization to satisfy $\Gamma(D) = 1$. The selected set S_0 must satisfy $C(S_0) \le B$, where $C(S_0)$ reserves three executions for each of the first two budget-feasible anchors before another block is admitted. Seeded random values break exact initialization ties, after which IDs provide deterministic candidate ordering. Adaptive candidate scoring uses 32 simulation draws, robust-family weight $\omega = 0.25$, coverage weight $\lambda = 0.05$, semantic reliability $\eta(a) = \max\{0.5, 1 - 0.25 s(a)\}$ for preregistered $s(a) \in [0, 1]$, and frozen relative cost $C(a)$. Every block may be executed at most three times total, so a first execution permits at most two later repeats. Repeats reduce the robust uncertainty contribution by $1/\sqrt{1 + n_t(a)}$ and are interleaved with new blocks.

More explicitly, $\widetilde{D}_a$ denotes a posterior-predictive complete block for action a, and

$$\Delta_t^R(a) = \left[\mathcal{R}(D_t) - \mathbb{E}_{\widetilde{D}_a \mid D_t} \mathcal{R}(D_t \cup \widetilde{D}_a)\right]_+ .$$

The expectation is approximated with the 32 frozen simulation draws. If $\mathcal{Q}(a)$ and $g(a)$ are the contrasts and task family touched by a, its repeat-adjusted robust uncertainty is

$$U_t(a) = \frac{\max_{q \in \mathcal{Q}(a)} \mathrm{SD}(\theta_{q,\mathrm{task},g(a)} \mid D_t)}{\sqrt{1 + n_t(a)}}.$$

For coverage type j among track, factor, semantic core, and task family, define $\gamma_j(D) = \min\{N_j(D)/r_j, 1\}$ with $r = (2, 14, 8, 2)$. Using uniform $\alpha_j = 1/4$, the soft acquisition gain and hard stopping gate are

$$G_t(a) = \sum_j \alpha_j \left[\gamma_j(D_t \cup a) - \gamma_j(D_t)\right],$$
$$\Gamma(D) = \min_j \gamma_j(D).$$

Thus, $G_t(a)$ rewards incremental breadth, whereas $\Gamma(D) = 1$ is required before confidence-based early stopping. The complete score is

$$\mathrm{score}_t(a) = \frac{\eta(a)\left[\Delta_t^R(a) + \omega U_t(a) + \lambda G_t(a)\right]}{C(a)}.$$

With remaining selection budget $b_t = B - \sum_{s<t} C(a_s)$, only actions satisfying $C(a) \le b_t$ and $n_t(a) < 3$ enter the feasible set $\mathcal{A}_t$. If this set is empty, diagnosis stops with budget exhaustion or candidate exhaustion as its recorded reason.

Each action executes every cell and shared task in a new block or an interleaved repeat. Its frozen cost is deducted for every attempted block. Infrastructure failures receive at most two logged retries; if the resulting block remains incomplete, it cannot update the posterior. Model refusals, invalid calls, truncation, and execution errors remain scored evidence when the block is complete. The bank stores frozen relative token and execution charges, with

$$C(a) = c^{\mathrm{rel}}_{\mathrm{tok}}(a) + c^{\mathrm{rel}}_{\mathrm{exec}}(a).$$

Raw realized tokens and latency are logged separately and do not retroactively change which actions were budget-feasible.

Let c_R count consecutive posterior states for which $\mathcal{R}(D_t) \leq 0.02$, resetting to zero after a complete-block update above the threshold; incomplete attempts leave both the posterior and c_R unchanged. Stopping occurs at budget or candidate exhaustion or, after minimum coverage, when either (i) $c_R \geq 2$ or (ii) the best cost-normalized acquisition score is below 0.001. Minimum coverage requires both tracks, all 14 factors, at least eight semantic cores, and at least two task families. Coverage failure prevents an apparently confident but narrow posterior from stopping. An implementation guard stops after 1,000 attempted actions, although the three-execution cap and finite Bank normally exhaust candidates first; this guard has stop reason `max_actions` and is not treated as convergence. Formally, if $\mathcal{X}_t, \mathcal{F}_t, \mathcal{Z}_t, \mathcal{G}_t$ are the tracks, factors, semantic cores, and task families represented by complete blocks, then

$$\Gamma(D_t) = \min\Bigg\{ \frac{|\mathcal{X}_t|}{2}, \frac{|\mathcal{F}_t|}{14}, \min\left(\frac{|\mathcal{Z}_t|}{8}, 1\right), \min\left(\frac{|\mathcal{G}_t|}{2}, 1\right) \Bigg\}.$$

The early-stopping coverage gate is met only when $\Gamma(D_t) = 1$; it is not a task-performance metric. These thresholds are denoted $\epsilon_R = 0.02$ and $\epsilon_A = 0.001$ in Algorithm 1 of the main paper. After stopping, the sentinel set is evaluated once under a fixed audit protocol. Sentinel outcomes never enter D_t, posterior updates, acquisition scores, or stopping decisions. Their cost is recorded in a separate audit ledger and included in total reported tokens identically for all compared methods.

C.4 Profile, Policy, and Diagnosis Artifacts

The diagnosis directory contains six artifacts:

`raw_runs.jsonl` requests, responses, validity, usage, deployment identity, and failures;

`paired_effects.jsonl` complete-block main and interaction contrasts;

`profile.json` posterior means, 95% credible intervals, ROPE probabilities, stability, heterogeneity, coverage, and decision regret;

`policy.json` per-contrast preferred levels, neutrality or DEFER, confidence, evidence counts, and source-independent conditions;

`report.md` a human-readable audit summary;

`manifest.json` counts, hashes, frozen metadata, budget, and stop reason.

Automated adaptation consumes the structured profile/policy and a generated `transformation_spec.json`. The policy is source-independent: it first records the recommended level, neutrality, or DEFER. The adaptation controller compares this recommendation with the source Skill fingerprint and derives KEEP, TRANSFORM, or DEFER; `report.md` is for human review and is never inserted into a rewrite prompt. The prompt contains the original skill, only approved transformations, confidence conditions, and invariants. Default invariants preserve semantic scope, safety constraints, output contracts, tools, permissions, and required fields. Supported mechanical transformations may be performed deterministically; structural or semantic-preserving rewrites may use a fixed stronger generator. Every candidate passes structure, semantic, non-inferiority, validation, and blind-test gates. The source is immutable, two failed repairs trigger rollback, and heterogeneous or weak evidence yields DEFER.

Artifact	Minimum content	Consumer and boundary
`raw_runs.jsonl`	Deployment hash, probe/variant/task/core IDs, seed, order, repeat, output, validity, utilities, typed failure, tokens, and latency	Audit and replay; append-only evidence.
`paired_effects.jsonl`	Complete-block IDs, factor or interaction contrast, channel, signed effect, family, seed, repeat, and deployment hash	Posterior fitting; incomplete or mixed-deployment blocks are absent.
`profile.json`	Per-contrast/family posterior moments, 95% interval, ROPE probabilities, Bayes action, stability, repeat consistency, heterogeneity, evidence counts, coverage, and regret	Diagnosis and drift comparison; never a rewrite prompt by itself.
`policy.json`	Preferred/disfavored configurations, provisional eligibility, confidence, effect, evidence, thresholds, and defer conditions	Source-independent controller input.
`report.md`	Decisions, confidence, effects, failures, use summary, and deployment label	Human audit only; excluded from model prompts.
`manifest.json`	Schema version, hashes, run/effect counts, failure summary, deployment metadata, budget ledger, sentinel ledger, and stop reason	Integrity, provenance, and cost accounting.

Table 10: Diagnosis artifact contract. Fields listed here are minimum fields; schemas permit additional evidence fields only where explicitly declared.

More precisely, the policy's `action` field denotes whether a contrast is eligible to drive a transformation; it is not yet an instruction to edit a specific source Skill. The fingerprint parser records detected levels and its confidence for all instruction and executable factors. For an eligible preference, an already matching source becomes KEEP, a confidently detected mismatch becomes TRANSFORM, and an absent or ambiguous source level becomes DEFER. Interaction recommendations are applied jointly: the controller does not transform one cell of a preferred interaction while silently ignoring the other.

The resulting transformation specification contains the immutable source hash, factor IDs, current and target levels, posterior confidence, approved operations, and invariants. Deterministic mode supports only registered mechanical rewrites. Prompt mode emits the specification and source without creating a candidate. LLM mode sends that same structured input to the frozen generator. Candidate-producing modes require independent structure, semantic, task, non-inferiority, and blind-test commands; each command consumes and emits one JSON object without a shell and must return `passed`. At most two repair attempts receive validation feedback. Failure after the second repair restores the source bytes and records the failed checks.

D Evaluation Details

D.1 Fair Comparisons

Diagnosis baselines receive the same probe bank, block indivisibility, cost accounting, seeds, and maximum actions. Migration baselines use identical source skill hashes, target model, generator when applicable, validation budget, and blind final tests. The "+**SkillProbe**" rows add profile conditioning to the native method rather than replacing its policy learning, memory, or optimizer. Published numbers obtained under different backbones or budgets are background only. The three family-reference rows in Table 12 are reported for scale and are excluded from the main table, best/second-best marks, and average-rank claims.

ALFWorld reports success for six task types and their aggregate; WebShop reports score and success; SkillsBench and SWE-Skills-Bench report pass rate. Search QA uses exact match with a frozen top-5 retriever and at most five search calls. Only NQ and HotpotQA provide skill-construction examples; the other five QA sets are out of domain. Diagnosis, adaptation/generation, validation, repair, and target execution costs are charged. Blind final-test feedback is never exposed to selectors or generators.

Matched diagnosis wrappers. Every diagnosis baseline selects an indivisible matched block; no method may query a favorable cell without paying for the remaining cells. Random samples uniformly from currently feasible blocks. Fixed D-optimal executes one frozen cost-adjusted design without outcome-dependent updates. Bayesian EIG maximizes expected posterior information gain, whereas the cost-aware IRT/OED wrapper divides its multidimensional item-information objective by $C(a)$. The tinyBench IRT++ wrapper estimates factor-specific latent traits and chooses blocks by

posterior score variance; Fluid updates an online item-response model and targets aggregate score error; continuous CAT models the continuous block score and uses its native confidence criterion. All wrappers receive the same repeat cap, initialization budget, failure records, and coverage accounting. Only **SKILLPROBE** optimizes terminal representation regret and explicitly includes the worst touched family and interaction coverage.

Matched downstream wrappers. For a "**+SKILLPROBE**" condition, the native method receives the same training data and optimization budget as its unconditioned row plus the structured representation directives from `policy.json`. It does not receive Bank answers, sentinel outcomes, `report.md`, or blind-test feedback. Generic rewriting, GEPA, SkillReducer, SkillRL, and SkillGraph share the same immutable source hash, target deployment, candidate generator where applicable, five validation gates, and two-repair limit. Native-budget results and results copied from another backbone are contextual only and are excluded from rankings.

D.2 RESEARCH QUESTIONS, MODELS, AND BUDGETS

The evaluation addresses five research questions. **RQ1** asks whether semantically equivalent Skill representations induce distinct, reproducible preferences across model deployments. **RQ2** asks whether **SKILLPROBE** can recover an actionable preference profile under a limited probe budget. **RQ3** tests whether profile conditioning improves cross-model Skill migration and complements strong adaptation baselines. **RQ4** tests whether the same profile interface supports applications beyond migration. **RQ5** identifies the components responsible for diagnostic quality, downstream utility, and efficiency.

For the preference-fingerprint analysis, we profile Qwen3.5-2B/9B/27B/35B-A3B Qwen Team (2026a), GPT-4o, Claude 3.7 Sonnet, and Gemini 2.5 Pro. The primary migration transfer is Qwen3.5-27B→Qwen3.5-9B (Table 2). Seven additional transfers (Table 3) cover Gemma 2 27B→9B, Llama 3.1 70B→8B, reciprocal Qwen3.5–Gemma 2 and Qwen3.5–Llama 3.1 directions, and Qwen3.5-27B→GPT-4o. Cold start holds out Qwen3.5-4B; drift uses Qwen3.5-27B→Qwen3.6-27B Qwen Team (2026b). A probe-run is one variant–task call. Selectable-pool budgets are 10%, 20%, and 40% of expected full-bank cost, with 20% predeclared as primary and 100% used as reference. Reported values average seeds 11, 23, and 47.

Open Qwen deployments use an OpenAI-compatible provider, temperature zero, and a 4,096-token output cap. GPT-4o and Gemini 2.5 Pro use temperature zero with the same cap; Claude 3.7 Sonnet uses its fixed medium reasoning setting and the same cap. Provider aliases are not treated as immutable revisions. Exact model revision, access timestamp, context limit, system-prompt hash, scaffold, container or environment digest, hardware, driver, and package versions are required in each final run manifest. A configuration with a null revision, access time, or environment digest is an executable template, not evidence for a reported deployment.

Diagnosis metrics are full-profile error, actionable decision agreement, sentinel direction accuracy, coverage, and realized tokens. The fixed post-diagnosis sentinel audit never affects selection or posterior fitting; its cost is recorded separately and included in total tokens for every method. All diagnosis, adaptation, generation, visible validation, repair, and target execution costs are charged. Final tests remain blind. For search QA, retrieval is fixed to top-5 and each example permits at most five search calls.

Reporting metrics. Let $\widetilde{\vartheta}_{m,q} = \vartheta_{m,q}/s_q$ be the signed aggregate task-quality effect after division by the frozen contrast scale s_q from the full-bank calibration manifest. The RQ1 heatmap reports these values. Its pairwise profile distance is the root-mean-square distance over the shared standardized contrasts,

$$d(m, m') = \left[|\mathcal{Q}|^{-1} \sum_{q \in \mathcal{Q}} (\widetilde{\vartheta}_{m,q} - \widetilde{\vartheta}_{m',q})^2 \right]^{1/2}.$$

Leave-one-core-out identification removes one semantic core from every model, forms a per-model centroid from the remaining core-level effect vectors, and assigns each held-out vector to the nearest Euclidean centroid. Rows of the confusion matrix are normalized over held-out predictions. Thus neither $d(m, m')$ nor identification accuracy is downstream task performance.

For RQ2, profile error is the normalized discrepancy between a budgeted profile and the independently replayed exhaustive reference profile; actionable decision agreement is the percentage of reference contrasts for which the gated level/neutral decision agrees; and sentinel direction accuracy is the percentage of held-out sentinel contrasts whose signed direction agrees with the frozen reference. The exhaustive method is itself estimated from finite runs, so its row need not have zero error or 100% agreement. Coverage is $100\Gamma(D)$ and reported tokens include the fixed sentinel audit. The run-level release must retain the reference IDs and scaling constants used to compute these aggregates; they cannot be reconstructed from the mean-only workbook.

For migration, average rank is the arithmetic mean of ranks on ALFWorld aggregate success, WebShop success, SkillsBench pass rate, and SWE-Skills-Bench pass rate; raw scores from unlike benchmarks are never averaged. For version drift, with no-adaptation performance P_0 and full-reevaluation performance P_{full}, recovery is

$$100\,\frac{P - P_0}{P_{\text{full}} - P_0}.$$

Reevaluation cost is the evaluated-skill cost relative to full reevaluation, and localization is recall of the affected-Skill set defined by full reevaluation. Drift for contrast q is declared only when $|\mu_q^{\text{new}} - \mu_q^{\text{old}}|$ exceeds both 0.03 and 1.96 times the joint posterior standard deviation.

Statistical boundary. The tables in this appendix contain only final means across the three seeds. The implementation supports a 10,000-draw paired hierarchical bootstrap over the preregistered cluster unit (semantic core for diagnosis, Skill for migration and generation, and task family for family-level summaries) and Holm correction for a declared comparison family. Because the aggregate workbook does not contain seed- or instance-level records, intervals and adjusted tests are not recoverable from it and no significance claim is made here. Such analyses may be added only from released append-only run records, without changing the reported means or selecting tests after seeing their outcomes.

Migration uses ALFWorld Shridhar et al. (2021), WebShop Yao et al. (2022), SkillsBench Li et al. (2026a), and SWE-Skills-Bench Han et al. (2026). Search QA uses single-hop NQ, TriviaQA, and PopQA Kwiatkowski et al. (2019); Joshi et al. (2017); Mallen et al. (2023), and multi-hop HotpotQA, 2Wiki, MuSiQue, and Bamboogle Yang et al. (2018); Ho et al. (2020); Trivedi et al. (2022); Press et al. (2023). Only NQ and HotpotQA provide Skill-construction data.

External benchmarks remain in their official repositories and are invoked by JSON-stdin/stdout adapters without a shell; interactive environments additionally pass an opaque state token between reset and step. Each final run manifest must record repository commit or release, split, selected example IDs and count, environment image, scorer version, and license or access note. QA manifests also record corpus and index hashes, retriever and reranker configuration, top-k, and search-call limit. These fields are intentionally not inferred from dataset names or mutable default branches.

Diagnosis compares Random, fixed D-optimal design, Bayesian EIG, tinyBench IRT++ Polo et al. (2024), Fluid Hofmann et al. (2025), cost-aware IRT/OED, and continuous CAT Balkir et al. (2026). Migration compares no Skill, ReAct, GRPO Shao et al. (2024), SimpleMem+GRPO, SkillRL, SkillGraph, generic rewriting, GEPA, and SkillReducer, together with their applicable profile-conditioned variants. Search QA includes RAG, Search-o1, Search-R1, ZeroSearch, StepSearch Lewis et al. (2020); Jin et al. (2025); Sun et al. (2025); Zheng et al. (2025), EvolveR, SkillRL, SkillGraph, and IGRPO. Main comparisons match budgets, source hashes, generator, validators, and blind tests; native configurations are reported separately and never mixed into the primary ranking.

D.3 Representation-Preference Results (RQ1)

Figure 1 in the main paper displays the preference effects, profile distances, and held-out identification matrix. Beyond the Qwen contrasts discussed there, Claude 3.7 Sonnet has its largest effects on validation (T7), constraint handling (T6), and recovery (E5), whereas Gemini 2.5 Pro peaks on examples (T8) and interactions IX2/IX4. Pairwise profile distances across all seven deployments range from 0.076 to 0.735.

In the leave-one-semantic-core-out classifier, diagonal identification rates range from 64% to 89% around the 75.7% macro accuracy reported in the main paper. Confusions concentrate among Qwen

variants, especially 9B and 35B-A3B, rather than among GPT-4o, Claude 3.7 Sonnet, and Gemini 2.5 Pro. This is evidence of a reproducible representation fingerprint, not a claim that probes generally recover model identity: the classifier sees only held-out preference effects. The held-out confusion pattern helps bound what the preference profile can identify beyond its training cores.

D.4 BUDGETED DIAGNOSIS RESULTS (RQ2)

Figure 3 and Table 4 in the main paper give the budget curves and primary 20% comparison. The additional budget points clarify how quickly the diagnosis approaches the full-bank reference.

Complete baseline metrics at the primary budget are in Table 11.

Method	Bud.	Err.↓	Agr.↑	Sent.↑	Cov.↑	Tok.↓
Full Bank	100	6.1	98.8	97.4	100.0	1280
Random	20	44.2	41.6	69.4	63.2	254
Fixed D-opt.	20	<u>30.5</u>	53.8	75.2	70.8	247
Bayesian EIG	20	39.8	49.3	81.6	77.6	266
tinyBench IRT++	20	33.3	50.2	78.1	74.1	238
Fluid	20	48.9	42.5	82.4	80.3	272
Cost-aware IRT/OED	20	39.3	49.6	84.2	82.5	<u>221</u>
Continuous CAT	20	<u>30.5</u>	<u>67.3</u>	<u>85.6</u>	<u>84.0</u>	244
SKILLPROBE	20	**11.9**	**84.3**	**92.9**	**93.7**	**198**

Table 11: Diagnosis at the primary 20% budget. Full Bank denotes exhaustive 100% probing, not an error-free oracle; its residual error reflects finite-sample and evaluation noise.

The advantage is largest under tight budgets. At 10%, **SKILLPROBE** reaches 69.4% decision agreement and 85.9% sentinel accuracy, compared with 52.7% and 75.6% for continuous CAT. At 40%, it reaches 95.1% agreement, 95.8% sentinel accuracy, and profile error 7.8, already close to Full Bank's 98.8%, 97.4%, and 6.1. All methods converge near 100%, arguing against a scorer-specific ceiling as the explanation. The primary run consumes 198k tokens, 15.5% of Full Bank's 1,280k. The nominal 20% budget is an expected-cost upper bound; cost-aware acquisition and early stopping leave part of it unspent.

The baseline ordering clarifies the gain. D-optimal selection improves coverage over random selection but does not target a terminal representation decision. EIG can prioritize posterior uncertainty that cannot change an action. Continuous CAT is stronger, but lacks paired semantic blocks, interaction coverage, and the worst-family regret term. These differences explain why its budget curve trails the decision-aware policy. Sentinel accuracy remains below Full Bank, so the resulting profile is confidence-gated rather than treated as complete.

D.5 COMPLETE MIGRATION RESULTS

Group	Method	ALFWorld							WebShop		SkillsBench		SWE-Skills		Rank
		Pick	Look	Clean	Heat	Cool	Pick2	All	Score	Succ.	Pass	Δ	Pass	Δ	↓
Reference	GPT-family reference	75.3	60.8	31.2	56.7	21.6	49.8	48.0	31.8	23.7	57.8	–	42.0	–	10.75
Reference	Claude-family reference	88.1	75.3	70.7	78.0	55.4	68.6	72.3	48.5	41.2	62.4	–	48.5	–	6.00
Reference	Gemini-family reference	92.8	63.3	62.1	69.0	26.6	58.7	60.3	42.5	35.9	55.6	–	41.8	–	10.50
Backbone	Qwen3.5-9B (No Skill)	42.3	31.7	28.3	20.2	16.8	13.1	25.9	33.0	12.5	36.4	0.0	28.6	0.0	19.25
Prompt/Memory	ReAct	53.2	39.6	38.2	24.8	26.4	23.7	35.9	48.7	21.5	39.0	2.6	30.1	1.5	17.25
Prompt/Memory	Reflexion	65.8	45.6	48.9	36.4	41.8	28.4	46.0	60.3	31.4	41.5	5.1	31.0	2.4	15.50
RL	GRPO	91.6	68.0	90.5	76.0	74.4	66.9	79.3	80.6	67.8	44.1	7.7	34.9	6.3	11.00
Memory/RL	EvolveR	68.0	38.5	50.3	20.3	37.6	36.6	47.8	45.2	20.3	42.3	5.9	33.3	4.7	15.50
Memory/RL	SimpleMem+GRPO	90.0	64.7	62.5	53.6	66.6	29.7	64.1	69.1	49.0	45.8	9.4	35.3	6.7	12.25
Skill/RL	SkillRL	97.0	72.8	91.4	90.6	95.2	87.6	90.2	86.0	73.8	52.4	16.0	40.4	11.8	7.75
Skill/RL	SkillGraph	**99.8**	80.0	98.1	98.7	82.7	85.4	91.1	**92.1**	**84.7**	54.9	18.5	41.1	12.5	5.25
Migration	Original Skill	46.2	34.8	30.1	18.0	14.6	12.3	25.8	34.5	12.8	35.7	-0.7	27.4	-1.2	19.75
Migration	Generic LLM Rewrite	52.7	38.1	34.6	25.0	22.7	18.3	31.0	39.1	16.1	39.4	3.0	30.7	2.1	17.50
Migration	GEPA	88.7	67.2	84.7	72.0	70.9	61.3	74.2	78.5	64.4	48.7	12.3	36.8	8.2	10.50
Migration	SkillReducer	84.3	64.2	80.3	68.2	66.1	58.4	70.3	75.2	60.3	49.3	12.9	38.9	10.3	10.75
Migration	SkillProbe Migration	96.3	76.3	94.9	92.6	93.8	86.9	90.0	88.4	78.4	58.8	22.4	45.5	16.9	5.25
Combination	GEPA + SkillProbe	97.4	77.4	95.8	93.6	94.7	86.7	90.0	89.6	79.9	59.4	23.0	45.1	16.5	5.00
Combination	SkillReducer + SkillProbe	96.5	75.7	94.4	92.4	92.4	85.3	89.1	88.6	77.8	60.7	24.3	46.7	18.1	5.25
Combination	SkillRL + SkillProbe	98.1	78.3	95.7	95.5	**96.0**	**88.5**	91.0	90.6	82.7	61.1	24.7	**47.5**	**18.9**	2.75
Combination	SkillGraph + SkillProbe	99.1	**81.5**	**99.4**	**99.8**	83.6	84.1	**91.5**	92.0	84.5	**62.4**	**26.0**	**47.5**	**18.9**	**1.50**

Table 12: Complete migration workbook for Qwen3.5-27B→Qwen3.5-9B. Family references use different deployments and are contextual only. Δ is relative to Qwen3.5-9B without a skill.

Expanded migration analysis (RQ3). Table 12 shows that a 27B source Skill slightly hurts 9B: ALFWorld aggregate success is 25.8 versus 25.9 without a Skill, and SWE-Skills falls from 28.6 to 27.4. **SKILLPROBE** migration instead reaches 90.0 ALFWorld, 78.4 WebShop success, 58.8 SkillsBench, and 45.5 SWE-Skills, tying base SkillGraph's average rank without replacing policy training.

Adding **SKILLPROBE** to GEPA, SkillReducer, SkillRL, and SkillGraph improves SkillsBench by 10.7/11.4/8.7/7.5 points and SWE-Skills by 8.3/7.8/7.1/6.4 points. SkillGraph+**SKILLPROBE** ranks first (1.50), followed by SkillRL+**SKILLPROBE** (2.75). However, SkillGraph+**SKILLPROBE** is 0.2 below SkillGraph on WebShop success and 0.1 below it on WebShop score. This saturation case rules out a universal-improvement interpretation.

The largest gains occur on SkillsBench and SWE-Skills-Bench, where procedural structure, validation, and recovery directly affect execution. Gains on ALFWorld and WebShop are smaller for already optimized SkillRL/SkillGraph policies, but the combined methods preserve strong interactive-agent performance while improving cross-domain rank. The original-Skill row is an important control: semantic relevance alone does not ensure representation compatibility with the target deployment. These results support RQ3, with gains narrowing when the native method is already near saturation.

D.6 COMPLETE SEARCH-QA RESULTS

Expanded search-QA analysis. Table 13 shows that SkillGraph+**SKILLPROBE** leads macro exact match at 53.0 (+2.3 over SkillGraph), SkillRL+**SKILLPROBE** gains 2.6, and EvolveR+**SKILLPROBE** gains 6.5. Gains extend across all five out-of-domain datasets, but ZeroSearch remains best on PopQA and StepSearch remains best on MuSiQue. The result therefore demonstrates broad complementarity rather than universal dominance over search reinforcement learning.

Because only NQ and HotpotQA contribute Skill-construction examples, improvements on TriviaQA, PopQA, 2Wiki, MuSiQue, and Bamboogle measure representation transfer rather than benchmark-specific rewriting. The strongest combined method improves both in-domain columns and four of five out-of-domain column maxima, although specialized search training still wins two datasets. The profile is therefore an orthogonal prior over how a search Skill should be expressed, not a replacement for retrieval or search-policy learning.

Group	Method	NQ†	TriviaQA∗	PopQA∗	HotpotQA†	2Wiki∗	MuSiQue∗	Bamboogle∗	Avg.
Backbone	Qwen3.5 Base	14.6	38.2	3.5	19.7	25.8	6.3	17.6	18.0
Prompt	CoT	16.4	39.5	5.5	20.3	26.4	8.7	26.7	20.5
Retrieval	RAG	30.8	60.9	20.7	28.4	25.6	11.2	19.0	28.1
Search/RL	Search-o1	22.7	43.8	14.5	20.8	29.6	10.5	33.8	25.1
Search/RL	R1-Instruct	24.0	47.3	19.7	24.9	30.3	8.0	22.5	25.2
Search/RL	Search-R1	42.7	63.0	42.3	39.2	42.9	16.7	39.9	41.0
Search/RL	ZeroSearch	46.4	63.8	**54.2**	37.7	37.0	20.4	30.6	41.4
Search/RL	StepSearch	40.5	60.8	38.7	41.8	39.9	**25.2**	43.6	41.5
Evolution	EvolveR	46.4	65.8	47.2	40.9	44.3	17.3	57.4	45.6
Skill	SkillRL	48.1	65.0	48.6	45.6	42.7	22.3	75.8	49.7
Skill	SkillGraph	50.1	65.5	50.7	46.4	46.7	21.2	74.4	50.7
Search/RL	IGRPO	47.5	64.1	49.1	44.3	45.4	23.5	68.4	48.9
Migration	Original Search Skill	39.1	59.3	40.9	35.8	37.7	15.4	42.1	38.6
Migration	Generic LLM Rewrite	43.5	61.5	44.6	39.9	40.4	17.8	50.7	42.6
Migration	SkillProbe Migration	50.4	65.0	51.4	46.9	45.6	24.3	76.8	51.5
Combination	EvolveR + SkillProbe	51.8	66.0	52.2	47.7	46.0	24.9	76.1	52.1
Combination	SkillRL + SkillProbe	51.4	66.6	52.3	47.5	46.3	24.0	**77.9**	52.3
Combination	SkillGraph + SkillProbe	**52.7**	**66.8**	53.8	**48.4**	**47.3**	24.6	77.1	**53.0**

Table 13: Complete search-augmented QA workbook. † denotes in-domain skill construction and ∗ out-of-domain evaluation.

D.7 Application Values and Extended Analysis (RQ4)

Figure 4 in the main paper summarizes cold start, skill generation, and version-drift repair. Tables 14, 15, and 16 give the per-budget, per-round, and per-method values behind those panels. For example, the 20% cold-start score is 52.8 on SkillsBench, 8.3 points above the inherited profile; the tables also retain the intermediate generation rounds and drift controls needed to inspect these comparisons.

The four applications share the same structured profile but use independent downstream procedures. Cold start estimates a new deployment without inheriting another model's preferences; migration rewrites an existing Skill; generation conditions iterative proposal and validation; drift monitoring localizes changed factors before adapting only affected Skills. No downstream test data are pooled. The drift controls in Table 16 separate localization and recovery from the cost of reevaluation, making the application comparison auditable.

Application protocols. Cold start diagnoses the held-out deployment directly and converts only confidence-gated policy entries into representation directives; the inherited condition instead reuses another deployment's profile without target probes. Task semantics come from the downstream task, never from Bank cores. Full Bank is an upper reference using all selectable blocks.

Generation produces one candidate per round for five rounds. At round r, the frozen generator receives the task, actionable representation directives, the best development candidate so far, and validator feedback from round $r-1$; its seed is the experiment seed plus $r-1$. The best validated score is retained rather than automatically accepting the last candidate. Only development feedback is visible, and the complete generation/validation trajectory is charged to the budget ledger.

For drift, old and new profiles are compared contrast by contrast using the joint-uncertainty rule above. Missing contrasts are deferred. The changed factors are matched against stored Skill fingerprints to form an affected-Skill set, and only that set is reevaluated and repaired. Full reevaluation defines the affected-set reference and the 100% cost denominator; global rewriting receives no localization signal.

Budget	Inherited	Random	Fixed	**SkillProbe**	Full
0	44.5	34.8	34.8	34.8	56.8
10	44.5	38.2	40.5	46.0	56.8
20	44.5	42.7	45.2	52.8	56.8
40	44.5	46.8	50.3	55.9	56.8
100	44.5	56.8	56.8	56.8	56.8

Table 14: Cold-start SkillsBench pass rate on held-out Qwen3.5-4B.

Round	One-shot	Self-FB	Creator	GEPA	EvoSkills	SkillGen	**SKILLPROBE**
1	37.0	38.3	40.7	41.0	42.6	43.1	45.3
2	37.0	41.7	43.2	45.6	47.2	48.6	51.4
3	37.0	42.0	45.5	48.0	51.7	52.6	56.3
4	37.0	43.8	46.3	51.9	54.7	55.2	59.8
5	37.0	43.7	46.1	50.3	55.8	56.7	60.8

Table 15: SkillLearnBench performance across optimization rounds.

Method	Perf.	Recovery	Re-eval.	Localization
No adaptation	41.2	0.0	0.0	18.0
Full re-evaluation	58.5	100.0	100.0	100.0
Global rewrite	52.3	64.2	48.0	35.0
SkillOps	55.5	82.7	31.5	78.5
SKILLPROBE local	57.8	96.0	18.6	92.4

Table 16: Adaptation after Qwen3.5-27B→Qwen3.6-27B drift.

D.8 COMPLETE COMPONENT ABLATION RESULTS (RQ5)

Table 17 reports all ablations at the primary budget, including cost normalization and adaptive stopping.

Setting	Err.↓	Agr.↑	Migr.↑	Tok.↓
SKILLPROBE (full)	11.9	84.3	**92.4**	198
w/o Bayesian Profile	15.4	87.8	88.6	<u>191</u>
w/o Decision Regret	12.7	84.1	83.2	196
w/o Cost Normalization	**11.6**	<u>91.1</u>	91.8	262
w/o Interaction Probes	16.8	86.9	86.0	**171**
w/o Adaptive Stopping	<u>11.7</u>	**91.8**	<u>92.0</u>	284

Table 17: Ablation at 20% budget. The integrated target is migration quality under cost.