BEYOND SEMANTIC RELEVANCE: GROUNDING SKILL SELECTION IN MODEL-SPECIFIC EXECUTION EXPERIENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Reusable skills provide language model agents with procedures for reasoning and problem solving. Models differ in their ability to use skills effectively, with small and medium language models struggling to reliably identify which skills will improve their answers. A natural approach is to record the outcomes of skill execution and use the resulting credit to guide selection. Directly transferring credit between backbones can reduce answer quality relative to learning credit from the receiving model's own executions. Therefore, we propose Executor-Calibrated Skill Selection (ECSS), which grounds skill selection in the execution experience of each model. ECSS predicts the benefit of a skill for a question from past executions. It then uses the model's current skill preferences to learn a correction for reward differences not explained by these credit estimates. Independent calibration determines how strongly the correction influences selection for each backbone and task domain. Across multiple datasets and model backbones, ECSS outperforms agent self-selection overall.

1 INTRODUCTION

Large language models (LLMs) provide a general foundation for language understanding and reasoning. Building on these capabilities, LLM-based agents combine reasoning with tool use and environmental feedback to accomplish tasks (Yao et al., 2023b). To reuse problem-solving procedures across tasks, agents can organize procedural knowledge into skill libraries (Xia et al., 2026; Zhang et al., 2026; Bai et al., 2026). In the natural-language setting studied here, skills describe operations such as question decomposition, entity resolution, and answer verification. Given a task, the agent reads the available skill descriptions, selects a procedure, and incorporates its instructions into the context used to generate an answer. Skill use therefore involves both choosing a procedure and executing it.

The value of a skill depends on how the model uses it. Models differ in their ability to understand an instruction, choose among alternatives, and execute the resulting procedure. For small and medium language models, the challenge is to predict which available skill will improve an answer. Figure 1a shows that allowing the agent to choose a skill can reduce answer quality relative to answering without a skill. Access to useful procedures and the ability to select them are therefore distinct capabilities.

Figure 1b makes this distinction concrete. Asked when a person's son died, the agent chooses a procedure that searches for a date in the question and fails to answer. A skill that first identifies the person from context instead yields the correct answer, as does answering without a skill. The agent thus overlooks an effective procedure and selects one that does not help. Such failures motivate grounding selection in evidence of execution benefit.

Past executions offer a natural source of this evidence. Comparing a skill's outcome with the outcome obtained without it yields a signal of its benefit, which we call *skill credit*. Learning from these signals connects future choices to observed performance, consistent with work on agent adaptation from execution feedback (Li et al., 2026; Wang et al., 2026). Transferable credit would reduce the execution history needed for each model.

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.



b A failed skill choice

HotpotQA | Qwen3-4B-Instruct-2507

Question: When did John Poindexter's son die?

C8 Find a date in the question, then its event.

C21 Identify the person using context clues.

C26 Use each answer to look up the next fact.

Agent selects C8 | Wrong answer | F1 = 0

"Not specified in the evidence"

Best observed: C21 | F1 = 100

"July 1, 2012" (one of 17 tied-best skills)

c Cross-Backbone Credit Transfer

Qwen3-14B credit → Qwen3-4B* executor

Dataset	Own F1	Transfer F1	Δ F1
MuSiQue	53.49	48.81	-4.68
HotpotQA	80.25	77.97	-2.28
2Wiki	82.38	83.04	+0.66

Figure 1: **Skill selection and cross-backbone credit transfer.** (a) Agent self-selection sometimes lowers answer quality relative to using no skill. (b) An illustrative failure: the agent chooses a date-lookup skill and answers incorrectly, whereas an entity-resolution skill yields the correct answer. (c) Direct credit transfer lowers performance on two of three tasks.

Credit learned from another backbone produces worse selections on MuSiQue and HotpotQA but a small gain on 2Wiki in the comparison shown in Figure 1c. Skill value is executor-dependent; we therefore learn credit from the model that will use the skill.

Historical outcomes and current preferences convey different information. Credit predicts execution benefit, whereas preferences express which skill the model considers appropriate. Aligning them requires a reward-trained correction.

We propose **Executor-Calibrated Skill Selection (ECSS)** to connect these signals through execution outcomes. ECSS first predicts each skill's benefit for a question using the executor's own historical rollouts, with performance without a skill as the reference. It then learns a correction for reward differences that the credit predictor does not explain, using the model's current preferences as input. Separate calibration data determine how strongly this correction influences selection for each backbone and task domain. At inference time, ECSS selects one skill or no skill before the executor generates an answer.

Across the primary multi-hop QA evaluations, ECSS improves average answer quality over agent self-selection. Our contributions are:

- We characterize skill selection through its benefit to the executor and document failures of agent selection and direct credit transfer.
- We introduce ECSS, which combines credit from the executor's own history with a correction informed by semantic preferences and calibrated on separate questions.
- We evaluate ECSS across benchmarks and backbones, using matched controls to assess historical credit and semantic correction.

2 Related Work

Language Agents and Tool Use. Agents couple language-model reasoning with external actions (Yao et al., 2023b; Schick et al., 2023; Shen et al., 2023). Modular and multi-agent systems expand the available action space (Karpas et al., 2022; Wu et al., 2024; Li et al., 2023a); API-oriented work studies how to expose and select tools (Patil et al., 2023; Qin et al., 2024; Hao et al., 2023b; Li et al., 2023b). AgentBench and WebArena evaluate interactive agent behavior (Liu et al., 2024; Zhou et al., 2024b). SayCan grounds language compatibility in value functions for embodied skills

(Ichter et al., 2023). We measure the answer-level value of natural-language instructions for a fixed language-model executor.

Reasoning Procedures and Skill Learning. Reasoning procedures include chain-of-thought and self-consistency (Wei et al., 2022; Wang et al., 2023b), decomposition and self-asking (Zhou et al., 2023; Khot et al., 2023; Press et al., 2023), and retrieval-interleaved reasoning (Trivedi et al., 2023). Other work explores search, computation, and verification within a procedure (Yao et al., 2023a; Hao et al., 2023a; Chen et al., 2023; Madaan et al., 2023; Dhuliawala et al., 2023; Gou et al., 2024). Self-Discover composes reasoning modules, while DSPy optimizes language-model programs (Zhou et al., 2024a; Khattab et al., 2024). Voyager stores reusable executable skills (Wang et al., 2023a). SkillRL, SAPO, and SkillDAG adapt skill banks or policies from experience (Xia et al., 2026; Zhang et al., 2026; Bai et al., 2026). ECSS selects one instruction from a fixed library for a fixed executor.

Experience, Memory, and Feedback. Language-agent memory can retain and organize prior interactions (Park et al., 2023; Packer et al., 2023; Sumers et al., 2024; Xu et al., 2025). Reflexion and ExpeL use task feedback to inform later decisions (Shinn et al., 2023; Zhao et al., 2024). ReSQL uses execution errors for reasoning-data construction and model adaptation (Park et al., 2026); JitRL estimates advantages from retrieved experience and modulates token logits (Li et al., 2026). ASSAY attributes marginal skill effects under randomized co-occurrence (Wang et al., 2026). ECSS uses outcome-fitted credit to choose one instruction before answer generation; its residual correction and calibration are tested against credit-only selection.

Adaptive Retrieval and Routing. Retrieval-augmented QA combines retrieved evidence and generation (Karpukhin et al., 2020; Izacard & Grave, 2021; Lewis et al., 2020); later systems decide when to retrieve or correct retrieval (Asai et al., 2024; Jiang et al., 2023c; Jeong et al., 2024; Yan et al., 2024). Model routing and output ensembling use query-dependent predictions to choose a model or combine generated responses (Chen et al., 2024; Ong et al., 2025; Hu et al., 2024; Jiang et al., 2023b). AgentSquare optimizes modular agent designs (Shang et al., 2025). ECSS routes among instructions for a fixed executor using that executor's outcomes.

3 Problem Formulation

Let x be a question, M a fixed language-model executor, and $\mathcal{S} = \{s_1, \ldots, s_K\}$ a library of natural-language procedures. The selector observes x and the skill descriptions, then chooses one action from $\mathcal{A} = \{\varnothing\} \cup \mathcal{S}$, where $\varnothing$ denotes answering without a skill. The executor generates an answer using the chosen instruction and any supplied evidence. Evidence is available to answer generation, but not to selection.

Let $Y_M(x, a; \xi) \in [0, 1]$ be the answer reward under a fixed evaluator, where ξ denotes generation randomness. Supplied evidence is fixed for each question and omitted from the notation. We define the utility of an action relative to the executor's performance without a skill:

$$u_M(x, a) = \mathbb{E}_\xi[Y_M(x, a; \xi) - Y_M(x, \varnothing; \xi)], \qquad a^*_M(x) \in \arg\max_{a \in \mathcal{A}} u_M(x, a). \tag{1}$$

Subtracting the no-skill reward preserves the optimal action and gives $u_M(x, \varnothing) = 0$. Utility concerns the model executing the skill, rather than an intrinsic property of its description.

The executor's semantic preference $p_M(a \mid x, \mathcal{S})$ expresses which instruction it favors, whereas $u_M(x, a)$ measures expected execution benefit. We seek a selector that uses past outcomes and current preferences to maximize the reward of its chosen action.

4 Executor-Calibrated Skill Selection

ECSS scores candidates by $c(x, a) + \eta g(x, a)$: historical credit c estimates execution benefit, a learned correction g adjusts relative action values using current preferences, and calibration sets its strength η (Figure 2). We use disjoint question sets to estimate credit (D_{credit}), fit the correction (D_{value}), calibrate its strength (D_{cal}), and evaluate the resulting selector (D_{eval}). Each executor–domain pair has its own fitted parameters and calibration; we omit these indices below.

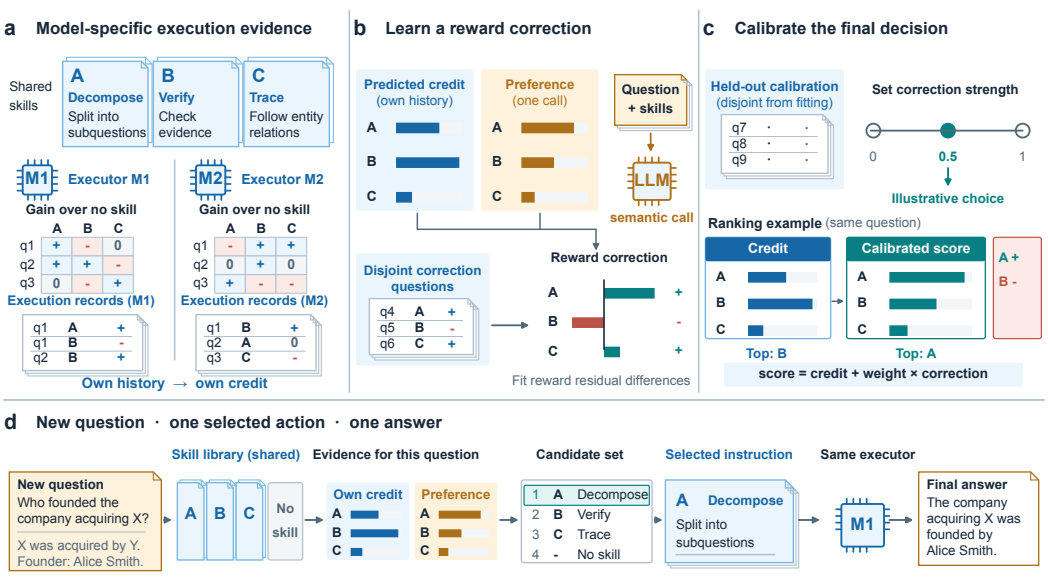

Figure 2: **Overview of Executor-Calibrated Skill Selection.** (a) Execution histories yield skill credit for each executor. (b) Credit and current preferences inform a correction fitted to observed reward contrasts on separate questions. (c) Held-out calibration sets the correction strength and can change the skill ranking. (d) The selected instruction is executed once to answer a new question. Signs, bars, rankings, and the worked question are schematic.

4.1 Executor-Specific Credit Estimation

We model skill utility as an average effect and a question-dependent deviation. On each question in D_{credit}, we execute no skill and a subset of skills sampled before observing rewards. Subtracting the question's recorded no-skill reward from each skill outcome provides a noisy utility label. The credit predictor is

$$c(x, a) = b_a + r(x, a), \qquad c(x, \varnothing) = 0, \tag{2}$$

where b_a is the empirical mean gain of skill a and $r(x, a)$ predicts the deviation for question x. Ridge regression fits the residual from frozen question–skill embedding features. We center the predicted residual for each skill over D_{credit}, so that b_a retains its mean interpretation. Labels and fitted parameters come from the executor's own outcomes; feature scaling and regularization use only credit-fitting data. Appendix A specifies the encoder, sampling scheme, and regression details.

4.2 Preference-Informed Reward Correction

To relate current preferences to execution benefit, we fit a correction to the reward differences left unexplained by credit.

Preference representation. A single selection call presents the question and full skill library, including no skill, to the executor. Normalizing its processed action-label log-probabilities gives $p_{xa} = p_M(a \mid x, \mathcal{S})$. Skill order and label mappings are shuffled, and retrieval scores are omitted. We represent each action with seven features:

$$\phi(x, a) = \left[b_a,\ r(x, a),\ p_{xa},\ R_{xa},\ I_{xa},\ p_{xa}H_x,\ I_{xa}G_x\right]^\top, \tag{3}$$

where R_{xa} is semantic rank normalized from best 1 to worst 0, I_{xa} indicates the most preferred action, H_x is normalized preference entropy, and G_x is the gap between the two highest credit scores. Credit features allow the correction to adjust historical predictions; semantic features capture preferences, their dispersion, and their interaction with credit separation. All features use the complete action set.

Candidate construction. To limit outcome collection, we construct a shortlist $U(x)$ before observing rewards. It contains the winning actions of a fixed family of credit–probability and credit–rank combinations, together with the semantic top action (Appendix A, Eq. 6). Each distinct candidate is executed once on correction-fitting and calibration questions. The same construction defines the available choices at inference; the correction is learned and applied within this shortlist.

Learning relative corrections. For each observed candidate, define the residual $z_{xa} = Y(x, a) - c(x, a)$. Then $z_{xa} - z_{xb}$ is the observed reward difference between two actions minus the difference already predicted by credit. The shared no-skill reference cancels, so raw rewards and gains relative to no skill yield the same contrast. We model this contrast with a linear correction $g(x, a) = \widetilde{\phi}(x, a)^\top \hat{w}$, where each coordinate of $\widetilde{\phi}$ is divided by the root-mean-square difference over fitting action pairs, with equal total weight per question.

Let $N = |D_{\text{value}}|$, $m_x = |U(x)|$, and $\Delta\widetilde{\phi}_{xab} = \widetilde{\phi}(x, a) - \widetilde{\phi}(x, b)$. We fit the correction by regularized least squares:

$$\hat{w} = \arg\min_{w} \left\{ \frac{1}{N} \sum_{\substack{x \\ |U(x)|\geq 2}} \frac{1}{\binom{m_x}{2}} \sum_{\{a,b\}\subset U(x)} \left[(z_{xa} - z_{xb}) - \Delta\widetilde{\phi}_{xab}^\top w \right]^2 + \lambda \|w\|_2^2 \right\}, \tag{4}$$

Questions with at least two candidates receive equal total weight; a singleton contributes zero loss and remains in N. The squared contrasts retain reward magnitudes and ties. The objective learns relative corrections, since action-independent offsets cancel from the loss. Ridge strength λ is selected within D_{value}; feature scales and fitted weights are then fixed. Appendix A gives an equivalent centered regression that avoids explicitly enumerating action pairs.

4.3 Decision Calibration and Inference

Regression fits reward differences, whereas selection depends on the highest-scoring action. We therefore calibrate the correction's effect on the decision itself. For strength η, the score and selection rule are

$$s_\eta(x, a) = c(x, a) + \eta g(x, a), \qquad \widehat{a}_\eta(x) = \arg\max_{a\in U(x)} s_\eta(x, a). \tag{5}$$

With both predictors fixed, we choose $\eta \in \{0, 0.5, 1\}$ to maximize the mean reward of $\widehat{a}_\eta(x)$ on D_{cal}. Zero retains the original credit $b+r$, one applies the full correction, and the intermediate value attenuates it. Exact calibration ties favor the smaller scale; action ties follow a fixed canonical order. The selected scale is specific to the executor and domain, while the features make the correction vary across questions and actions. Outer calibration leaves the fitted credit scale unchanged. We also report ECSS (credit only), which selects with $b + \rho r$ after calibrating ρ on D_{cal}. ECSS (score fusion) combines credit and semantic preferences.

For an unseen question, ECSS computes credit, obtains one semantic preference vector, constructs $U(x)$, and selects the highest calibrated score. The executor generates one answer using that action. The learned parameters and calibration remain fixed throughout evaluation. Online computation beyond the semantic call consists of embedding and linear scoring; offline outcome collection is a separate cost.

5 Experiments

We evaluate execution-grounded selection against direct agent choice and retrieval, then examine arithmetic transfer, component contributions, data requirements, and repeatability.

5.1 Experimental Setup

Tasks, backbones, and data. The primary evaluation covers answerable MuSiQue (Trivedi et al., 2022), HotpotQA (Yang et al., 2018), and 2Wiki (Ho et al., 2020), each with its own thirty-skill bank. Qwen3-4B-Instruct-2507 and Qwen3-14B (Yang et al., 2025) are evaluated on all three; Mistral-7B-Instruct-v0.3 (Jiang et al., 2023a) is evaluated on MuSiQue and HotpotQA. Each condition uses disjoint sets of 1,024 credit-fitting, 512 correction-fitting, 256 calibration, and 256 evaluation questions.

The primary comparison freezes the existing fits and reserves new calibration and evaluation questions. Backbones share question identities but collect their own rewards and fit separate parameters. All partitions are drawn from official training splits.

Execution and metrics. Answers receive the question, supplied context, and selected instructions, with a 768-token limit and temperature 0.2. We report normalized token F1 and exact match (EM) (Rajpurkar et al., 2016), maximizing over accepted aliases. Each selector produces one final answer. Identical question/model/action/prompt/seed answers are shared across selectors, not counted as independent samples. The arithmetic extension uses its own bank and answer-accuracy metric. Prompts, seeds, parsing and failure handling are specified in Appendices A and C.

Comparisons. *STOP* answers without a skill. *BGE Top1* selects by embedding similarity (Xiao et al., 2023); *same-bank SAPO Top3* concatenates three retrieved instructions, evaluating that selection rule rather than SAPO's full co-training system. *Agent* directly selects one action from shuffled skill descriptions in a single call, without historical rewards or retrieval ranks. These baselines and both ECSS variants use the same executor and task-specific bank. The credit-only variant was predeclared as a control for the full method; other component controls are evaluated in Section 5.4. Implementation details and the supplementary selection-interface control appear in Appendix A.

Statistical reporting. Scores use a 0–100 scale; bold marks the highest displayed mean. For absolute F1, Table 1 reports the equally weighted eight-condition mean and its question-bootstrap standard deviation (SD). For method differences, we report paired 95% bootstrap confidence intervals (CIs); ECSS minus Agent is the prespecified primary contrast. Both resample questions within each dataset with shared draws across backbones and condition on fixed fits and recorded answers. Answer-generation and calibration repeats instead report sample SD across runs (Appendix G).

5.2 Main Results

Table 1 compares both ECSS variants with four baselines on new questions from the same task domains, with fits and the learning recipe fixed. Calibration and evaluation questions are disjoint from one another and from the earlier development questions.

Table 1: **Main results on new questions.** Token F1 (↑). Credit only omits the learned preference correction; full uses it. 4B denotes Qwen3-4B-Instruct-2507; 14B, Qwen3-14B; and M7B, Mistral-7B-Instruct-v0.3. Avg. reports the equally weighted mean ± question-bootstrap SD; individual condition scores are point estimates.

	MuSiQue			HotpotQA			2Wiki		
Method	4B	14B	M7B	4B	14B	M7B	4B	14B	Avg.
No skill	53.45	48.49	21.18	73.74	70.51	38.14	78.04	66.00	56.20 ± 1.13
BGE Top1	53.95	**50.35**	21.39	79.74	70.74	36.52	82.47	66.97	57.77 ± 1.12
SAPO Top3	44.90	49.98	19.82	73.64	71.02	25.42	72.05	67.85	53.08 ± 1.07
Agent	51.72	49.72	22.16	72.78	72.88	35.96	**83.75**	67.31	57.04 ± 1.12
ECSS (credit only)	**54.49**	50.04	26.77	**81.47**	**75.36**	**44.46**	82.36	66.20	**60.14** ± 1.14
ECSS (full)	49.44	49.68	**27.26**	79.89	74.35	44.21	82.77	**68.58**	59.52 ± 1.16

Overall comparison. The credit-only and full ECSS variants achieve 60.14 and 59.52 mean F1, respectively, compared with 57.04 for direct Agent selection (Table 1). For the predeclared full-method contrast, the paired difference is 2.49 points, with a question-bootstrap 95% confidence interval of $[1.14, 3.85]$. The full variant also exceeds Top3 retrieval by 6.44 points and BGE Top1 by 1.76 points. Selection grounded in the executor's outcomes improves on direct choice and retrieval; the simpler credit-only variant has the higher observed mean.

Across models and tasks. The full ECSS variant exceeds Agent in five of eight conditions, with the largest gains on HotpotQA/4B and HotpotQA/Mistral. It trails Agent on MuSiQue/4B and 2Wiki/4B and by 0.04 points on MuSiQue/14B. The credit-only variant exceeds Agent in six of eight conditions. Section 5.4 examines the correction's behavior.

5.3 Extension to Arithmetic Reasoning

To test whether the approach extends beyond supplied-context question answering (QA), we apply the unchanged learning recipe to GSM8K (Cobbe et al., 2021). Each backbone collects its own arithmetic histories and fits new credit and correction models using a fixed thirty-skill arithmetic bank. No QA weights or rewards are transferred. Evaluation uses 256 held-out training questions per backbone and the upstream answer-accuracy metric (Appendix I); results appear in Table 2.

On Mistral, ECSS raises accuracy from Agent's 23.05 to 41.41, an 18.36-point gain; the paired-bootstrap 95% confidence interval runs from 11.33 to 25.39 points. It also exceeds Top3 by 16.02 points and no skill by 9.38 points. On 4B, where most selectors already approach 95%, ECSS reaches 95.31, 0.39 points above Agent. The gain concentrates in Mistral; 4B selectors already approach ceiling accuracy.

5.4 Ablation Study

Executor–domain calibration. Does correction strength need to depend on the executor and task? In a six-condition development ablation, we hold both fitted predictors fixed and change only how calibration evidence is pooled. Separate executor–domain calibration reaches 68.16 F1, versus 67.65 for a global scale (Table 3). The modest gain is concentrated on MuSiQue/4B: separate calibration disables the correction there, whereas pooled rules apply it fully. Full per-condition results and uncertainty appear in Appendix E.

Table 2: **GSM8K arithmetic results.** Answer accuracy (↑).

Method	Qwen3-4B	Mistral-7B
No skill	94.92	32.03
BGE Top1	94.92	20.70
SAPO Top3	92.97	25.39
Agent	94.92	23.05
ECSS (full)	**95.31**	**41.41**

Table 3: **Calibration pooling on development questions.** Token F1 and EM (↑).

Calibration rule	F1 ↑	EM ↑
Global	67.65	58.20
Executor only	67.65	58.20
Domain only	67.65	58.20
Executor–domain	**68.16**	**58.79**

Reward-target and feature controls. The separate six-condition development study in Table 4 tests the residual target, credit and value-label alignment, and semantic features. Removing semantic features lowers F1 on 2Wiki/14B but raises it on 2Wiki/4B; shuffling question-level credit also reverses its effect across tasks. These reversals show that semantic preference is not a uniformly useful correction to execution credit.

Table 4: **Reward-target and feature controls on development questions.** Token F1 (↑); Primary denotes ECSS.

Panel	Primary	Direct reward	Query shuffle	Label shuffle	Semantic zero
MuSiQue / 4B	53.72	54.24	51.40	53.72	53.72
MuSiQue / 14B	52.54	52.54	52.06	52.07	52.54
HotpotQA / 4B	77.83	77.00	78.44	76.97	77.37
HotpotQA / 14B	74.37	74.38	74.19	74.50	74.21
2Wiki / 4B	82.01	83.06	84.04	82.07	83.33
2Wiki / 14B	68.50	68.18	68.75	66.67	66.28

Correction-strength sensitivity. Our calibration grid spans the original credit ($\eta = 0$), an attenuated correction ($\eta = 0.5$), and the full fitted correction ($\eta = 1$). Figure 3 shows why the strength is calibrated rather than fixed: increasing η harms MuSiQue/4B but helps HotpotQA/14B, and the response differs across the six development conditions. The original three-point grid is a compact way to include both endpoints and an intermediate strength. A separate same-range grid comparison in Appendix F finds no mean F1 gain from denser search; the two-point grid nearly ties the three-point grid (Table 10).

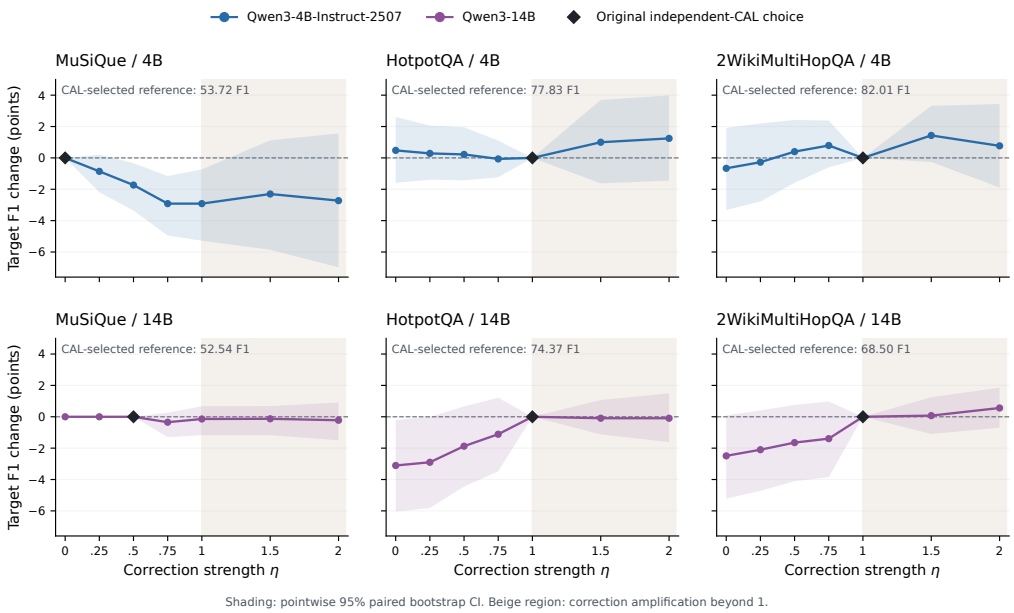

Figure 3: **Sensitivity to reward-correction strength.** Target token F1 changes relative to the original calibration-selected policy (diamonds) across six development conditions. Shading shows pointwise paired-question 95% intervals; $\eta > 1$ amplifies the correction beyond the deployed calibration range.

5.5 Data Requirements and Robustness

Value of additional execution history. The quality of execution history matters as well as its size. On MuSiQue/14B, fitting credit on 1,024 questions yields 54.58 F1, versus 51.96 with 128 questions and 50.75 when the full history's labels are shuffled. Intermediate budgets are nonmonotonic; 4B makes the same target choices across budgets (Table 20). The value of additional credit history therefore depends on the executor.

Stability under repeated calibration. With histories, fits, and target answers fixed, the full ECSS variant has lower F1 variation across three MuSiQue calibration sets than the credit-only variant: SD 0.11 versus 2.83 on 4B, and 0.19 versus 0.81 on 14B. Its mean F1 is lower in both conditions (Appendix G.3, Table 17). It also changes fewer selected skills across these sets (Appendix G.3). This measures calibration sensitivity, not end-to-end training or answer stability: a separate fixed-policy answer-stream repeat gives the full variant a higher SD than credit only (0.74 versus 0.16; Table 16).

Collection and deployment cost. A full credit history requires 8,192 answers per executor–domain condition; correction fitting and calibration add their candidate-action unions. Deployment uses one semantic call and one final answer, whereas credit only omits the semantic call. Appendix Table 10 measures offline CPU search time; end-to-end latency and equal-budget deployment utility have not been measured.

6 Discussion and Limitations

Execution history is the principal selection signal in our study. Both ECSS variants improve on direct agent choice in the primary QA cohort. The credit-only variant has the higher observed mean; the full variant shows less variation in decisions across the MuSiQue calibration samples. The correction combines credit and preference features, and the component controls do not establish a consistent aggregate contribution from preferences.

Credit source matters differently from correction design. In the separate matched-history development study, own-executor and pooled credit nearly tie before receiver calibration (58.51 versus 58.55 F1). With the same receiver-side calibration, own credit exceeds pooled credit (60.14 versus 58.57 F1) across eight conditions, though pooled credit remains better on 2Wiki/4B (Appendix D.1).

The evaluation uses fixed skill libraries and held-out questions from benchmark training splits. Shared articles, entities, and pretraining exposure remain possible across these questions. A QuALITY (Pang et al., 2022) development extension, completed before we focused on multi-hop QA, did not reproduce the gain: ECSS reached 64.06% accuracy, versus 65.62% for Agent and 68.75% for Top3. Backbone comparisons confound model size with training differences. Evaluation with evolving libraries and complete skill-learning agents would test whether the selection benefit persists when skill acquisition and policy adaptation interact.

Practical adoption depends on data collection cost and reliability. Each executor–domain pair requires an offline execution history, and ECSS adds a semantic call to credit-only selection. The repeated-calibration study holds histories and fits fixed, while the answer-generation repeat does not show lower ECSS variation. Independent recollection and refitting, compared with equally flexible credit-only controls at matched quality and total cost, would determine whether the correction justifies its additional machinery.

7 Conclusion

We introduced ECSS for skill selection in small and medium language models. ECSS estimates skill benefit from the executor's own outcomes and calibrates a learned correction informed by its current preferences. On held-out multi-hop QA questions, it improves average answer quality over direct agent selection, with gains in arithmetic reasoning that depend on the backbone.

References

Akari Asai, Zeqiu Wu, Yizhong Wang et al. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *International Conference on Learning Representations*, 2024.

Tong Bai, Zhenglin Wan, Pengfei Zhou et al. SkillDAG: Self-evolving typed skill graphs for LLM skill selection at scale. *arXiv preprint arXiv:2606.03056v1*, 2026.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*, 2024.

Wenhu Chen, Xueguang Ma, Xinyi Wang et al. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Transactions on Machine Learning Research*, 2023.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu et al. Chain-of-verification reduces hallucination in large language models. *arXiv preprint arXiv:2309.11495*, 2023.

Zhibin Gou, Zhihong Shao, Yeyun Gong et al. CRITIC: Large language models can self-correct with tool-interactive critiquing. In *International Conference on Learning Representations*, 2024.

Shibo Hao, Yi Gu, Haodi Ma et al. Reasoning with language model is planning with world model. In *Conference on Empirical Methods in Natural Language Processing*, pp. 8154–8173, 2023a.

Shibo Hao, Tianyang Liu, Zhen Wang et al. ToolkenGPT: Augmenting frozen language models with massive tools via tool embeddings. In *Advances in Neural Information Processing Systems 36*, 2023b.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara et al. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *International Conference on Computational Linguistics*, 2020.

Qitian Jason Hu, Jacob Bieker, Xiuyu Li et al. RouterBench: A benchmark for multi-LLM routing system. *arXiv preprint arXiv:2403.12031*, 2024.

Brian Ichter, Anthony Brohan, Yevgen Chebotar et al. Do as I can, not as I say: Grounding language in robotic affordances. In *Proceedings of the 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 287–318. PMLR, 2023.

Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open-domain question answering. In *Conference of the European Chapter of the Association for Computational Linguistics*, 2021.

Soyeong Jeong, Jinheon Baek, Sukmin Cho et al. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In *Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 7036–7050, 2024.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch et al. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023a.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. LLM-Blender: Ensembling large language models with pairwise ranking and generative fusion. In *Annual Meeting of the Association for Computational Linguistics*, 2023b.

Zhengbao Jiang, Frank F. Xu, Luyu Gao et al. Active retrieval augmented generation. In *Conference on Empirical Methods in Natural Language Processing*, 2023c.

Ehud Karpas, Omri Abend, Yonatan Belinkov et al. MRKL systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning. *arXiv preprint arXiv:2205.00445*, 2022.

Vladimir Karpukhin, Barlas Oguz, Sewon Min et al. Dense passage retrieval for open-domain question answering. In *Conference on Empirical Methods in Natural Language Processing*, 2020.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari et al. DSPy: Compiling declarative language model calls into state-of-the-art pipelines. In *International Conference on Learning Representations*, 2024.

Tushar Khot, Harsh Trivedi, Matthew Finlayson et al. Decomposed prompting: A modular approach for solving complex tasks. In *International Conference on Learning Representations*, 2023.

Patrick Lewis, Ethan Perez, Aleksandra Piktus et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems 33*, 2020.

Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani et al. CAMEL: Communicative agents for mind exploration of large language model society. *arXiv preprint arXiv:2303.17760*, 2023a.

Minghao Li, Yingxiu Zhao, Bowen Yu et al. API-bank: A comprehensive benchmark for tool-augmented LLMs. In *Conference on Empirical Methods in Natural Language Processing*, 2023b.

Yibo Li, Zijie Lin, Ailin Deng et al. Just-in-time reinforcement learning: Continual learning in LLM agents without gradient updates. *arXiv preprint arXiv:2601.18510v3*, 2026.

Xiao Liu, Hao Yu, Hanchen Zhang et al. AgentBench: Evaluating LLMs as agents. In *International Conference on Learning Representations*, 2024.

Aman Madaan, Niket Tandon, Prakhar Gupta et al. Self-Refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems 36*, 2023.

Isaac Ong, Amjad Almahairi, Vincent Wu et al. RouteLLM: Learning to route LLMs from preference data. In *International Conference on Learning Representations*, 2025.

Charles Packer, Sarah Wooders, Kevin Lin et al. MemGPT: Towards LLMs as operating systems. *arXiv preprint arXiv:2310.08560*, 2023.

Richard Yuanzhe Pang, Alicia Parrish, Nitish Joshi et al. QuALITY: Question answering with long input texts, yes! In *Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 5336–5358, 2022.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai et al. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, 2023.

Minjun Park, Yongju Seong, Myoseop Sim et al. ReSQL: Self-improving framework for reasoning-aware text-to-SQL dataset generation. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 33582–33602. Association for Computational Linguistics, 2026.

Shishir G. Patil, Tianjun Zhang, Xin Wang et al. Gorilla: Large language model connected with massive APIs. *arXiv preprint arXiv:2305.15334*, 2023.

Ofir Press, Muru Zhang, Sewon Min et al. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023.

Yujia Qin, Shihao Liang, Yining Ye et al. ToolLLM: Facilitating large language models to master 16000+ real-world APIs. In *International Conference on Learning Representations*, 2024.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev et al. SQuAD: 100,000+ questions for machine comprehension of text. In *Conference on Empirical Methods in Natural Language Processing*, 2016.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessi et al. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems 36*, 2023.

Yu Shang, Yu Li, Keyu Zhao et al. AgentSquare: Automatic LLM agent search in modular design space. In *International Conference on Learning Representations*, 2025.

Yongliang Shen, Kaitao Song, Xu Tan et al. HuggingGPT: Solving AI tasks with ChatGPT and its friends in Hugging Face. In *Advances in Neural Information Processing Systems 36*, 2023.

Noah Shinn, Federico Cassano, Ashwin Gopinath et al. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, 2023.

Theodore R. Sumers, Shunyu Yao, Karthik Narasimhan et al. Cognitive architectures for language agents. *Transactions on Machine Learning Research*, 2024.

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

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot et al. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Annual Meeting of the Association for Computational Linguistics*, 2023.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang et al. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*, 2023a.

Xuezhi Wang, Jason Wei, Dale Schuurmans et al. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023b.

Yixuan Wang, Yiyang Zhou, Yiming Liang et al. Not all skills help: Measuring and repairing agent knowledge. *arXiv preprint arXiv:2606.15390*, 2026.

Jason Wei, Xuezhi Wang, Dale Schuurmans et al. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35*, 2022.

Qingyun Wu, Gagan Bansal, Jieyu Zhang et al. AutoGen: Enabling next-gen LLM applications via multi-agent conversation. In *Conference on Language Modeling*, 2024.

Peng Xia, Jianwen Chen, Hanyang Wang et al. SkillRL: Evolving agents via recursive skill-augmented reinforcement learning. *arXiv preprint arXiv:2602.08234*, 2026.

Shitao Xiao, Zheng Liu, Peitian Zhang et al. C-Pack: Packed resources for general chinese embeddings. *arXiv preprint arXiv:2309.07597*, 2023.

Wujiang Xu, Zujie Liang, Kai Mei et al. A-MEM: Agentic memory for LLM agents. In *Advances in Neural Information Processing Systems 38*, 2025.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu et al. Corrective retrieval augmented generation. *arXiv preprint arXiv:2401.15884*, 2024.

An Yang, Anfeng Li, Baosong Yang et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Zhilin Yang, Peng Qi, Saizheng Zhang et al. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Conference on Empirical Methods in Natural Language Processing*, 2018.

Shunyu Yao, Dian Yu, Jeffrey Zhao et al. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems 36*, 2023a.

Shunyu Yao, Jeffrey Zhao, Dian Yu et al. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023b.

Zhiwei Zhang, Yudi Lin, Nikki Lijing Kuang et al. Co-evolving skill generation and policy optimization. *arXiv preprint arXiv:2606.08755*, 2026.

Andrew Zhao, Daniel Huang, Quentin Xu et al. ExpeL: LLM agents are experiential learners. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):19632–19642, 2024.

Denny Zhou, Nathanael Scharli, Le Hou et al. Least-to-most prompting enables complex reasoning in large language models. In *International Conference on Learning Representations*, 2023.

Pei Zhou, Jay Pujara, Xiang Ren et al. Self-Discover: Large language models self-compose reasoning structures. In *Advances in Neural Information Processing Systems 37*, 2024a.

Shuyan Zhou, Frank F. Xu, Hao Zhu et al. WebArena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations*, 2024b.

AI USE STATEMENT

AI assistants supported hypothesis refinement, experimental design, implementation, experiment orchestration, data preparation, interpretation of results, literature review, figure preparation, and manuscript drafting and editing. The accompanying artifacts include source-identity checks, raw-response replay, deterministic score checks, and tests of analysis code. These automated checks do not substitute for author review; the authors retain responsibility for the final submission and its claims.

A IMPLEMENTATION AND DATA SPECIFICATION

Skill banks and embeddings. Each dataset uses thirty frozen native skill cards, represented by title, principle, and applicability description. The same bank is used by all selectors and backbones within that dataset; each dataset has its own bank. Credit and retrieval features use `BAAI/bge-small-en-v1.5`, classification-token pooling, ℓ_2 normalization, and a 512-token encoding limit. For unit question and skill vectors q, v_a, the ridge features are $[q^\top v_a; q \odot v_a]$. The encoder is frozen and never sees reward labels. Credit feature standardization and skillwise residual centering use credit-fitting data only.

Credit fitting. Each credit-fitting question is answered without a skill and with seven skills sampled from the thirty-card bank before observing rewards. For each skill, b_a is its empirical mean observed reward improvement over the question-matched STOP answer. The answers need not share a generation seed. All thirty skills have fitting observations. Ridge strength is chosen from $\{1, 10, 100, 1000\}$ using five fitting-only folds, minimizing selected-action loss within the seven observed actions. Ties use lower prediction mean squared error and then larger regularization. The constant inclusion weight $30/7$ enters the residual ridge objective. STOP has exactly zero mean and residual. Credit is specific to the executor and generation protocol.

Correction features and fitting. For the seven features in Eq. 3, R is ordinal rank mapped from best 1 to worst 0; $I_{xa} = \mathbb{1}[a = \arg\max_b p_{xb}]$; $H_x = -\sum_a p_{xa} \log p_{xa} / \log 31$; and G_x is the top-two gap of $b + r$. Probabilities, entropy, top action, ranks, and gap use the full action space before support masking. Feature scales use training-only root-mean-square pair differences with floor 10^{-8}. There is no intercept, since it cancels from comparisons. Ridge strength is chosen from $\{0.01, 0.1, 1, 10\}$ by three inner folds on the correction-fitting split and selected-action reward; the value model is then refit on all 512 correction-fitting questions. The independent 256-question calibration selects the final η.

The pairwise objective has a linear-size implementation. For each question, center supported feature rows and residual rewards by their supported means. If $m_x \geq 2$, each centered row receives weight $2/[N(m_x - 1)]$ in the least-squares system. This is exactly the mean unordered-pair loss, since $\sum_{a<b}(t_a - t_b)^2 = m \sum_a (t_a - \bar{t})^2$. A singleton support contributes zero loss but remains in the query count. This identity avoids treating a question with many observed actions as many independent training examples.

Outcome-blind support. The prespecified score family used to form $U(x)$ is

$$b_a + \rho r(x, a) + \beta h(x, a), \qquad \rho \in \{0, 0.5, 1\}, \quad \beta \in \{0, 0.01, 0.03, 0.1, 0.3, 1\}, \tag{6}$$

where h is either semantic probability or its normalized ordinal rank. Each of the two eighteen-rule families also includes the semantic top choice. The union is fixed before collecting answer rewards; at deployment only the selected action is executed, not all candidates. Features use all 31 actions even though correction choices are restricted to this outcome-blind support.

Calibration and tie handling. Each learned correction has a separate $\eta \in \{0, 0.5, 1\}$, chosen by selected-action mean calibration reward. The lower η wins an exact tie. Pure credit selects $\rho \in \{0, 0.5, 1\}$, and raw fusion selects one of the eighteen probability-score configurations in Eq. 6 or the preference-argmax endpoint. Ordinary policy ties prefer fewer active components, then lower ρ, lower β, and canonical configuration name. Action-score ties follow canonical action IDs. No target answer is used in these choices. The ordinal configurations are used to construct support, not selected post hoc as an extra favorable target role.

Models and selection interfaces. The exact checkpoints are Qwen3-0.6B, Qwen3-4B-Instruct-2507, Qwen3-14B, Qwen3.5-35B-A3B, and Mistral-7B-Instruct-v0.3. Qwen thinking is disabled; Mistral uses its native non-thinking template. The semantic preference call has temperature zero and one generated label token with all 31 processed label log-probabilities available. Labels are verified as single tokens for the checkpoint, and their mappings to actions are shuffled. The supplementary preference-argmax control selects the highest-probability action. Agent instead has one 128-token JSON selection call. Neither sees evidence passages, correctness labels, similarity scores, or retrieval ordering in its selection prompt. The answer stage sees the evidence and actual selected skill text. Thus, the two semantic interfaces share information access but not output format or selection-token count.

Collection versus deployed cost. Each original development condition uses 1,024 credit-fitting questions and 8,192 associated answers, followed by outcome-blind supported actions on 512 correction-fitting and 256 calibration questions. Its twelve selectors are evaluated on 256 target questions, with identical ordered actions deduplicated. The primary new-question cohort instead has eight selectors. For the completed six-panel HotpotQA/2Wiki study, data collection required 79,867 answer executions and 7,680 formal selector calls across all stages, plus six synthetic interface probes. Deployment uses one selector call and one answer; the collection totals are separate from inference latency.

B Development Comparisons on Multi-Hop QA

Table 5 summarizes the original development cohort. Checkpoints share question identities within each dataset; credit comes from prior executions, and correction scales are selected on separate calibration questions. Complete per-condition F1 and exact-match exports remain in the source artifacts. The primary new-question evaluation uses a distinct cohort.

Table 5: **Development comparison across multi-hop QA tasks.** Token F1 (↑).

Backbone	Agent	Preference argmax	Top3	ECSS (credit only)	ECSS (score fusion)	ECSS (full)	η
MuSiQue							
Qwen3-0.6B	10.56	10.39	11.62	9.24	10.24	**12.08**	1
Qwen3-4B	49.76	51.52	44.48	52.87	52.74	**53.72**	0
Qwen3-14B	51.42	50.84	**53.70**	51.85	50.43	52.54	0.5
Qwen3.5-35B-A3B	62.80	62.00	65.03	**67.10**	**67.10**	66.19	1
Mistral-7B	25.38	24.21	18.11	26.31	26.09	**26.45**	1
HotpotQA							
Qwen3-4B	73.87	74.62	74.20	77.99	**80.11**	77.83	1
Qwen3-14B	72.74	72.83	67.67	74.21	**75.43**	74.37	1
Qwen3.5-35B-A3B	85.39	86.66	**87.24**	86.95	86.95	86.99	0.5
Mistral-7B	39.02	38.99	25.07	**43.81**	41.86	42.74	1
2Wiki							
Qwen3-4B	81.74	**83.64**	71.86	82.92	83.49	82.01	1
Qwen3-14B	67.60	68.18	67.94	67.46	66.84	**68.50**	1
Qwen3.5-35B-A3B	91.02	**92.47**	91.39	92.16	92.16	91.38	0.5

C Execution Incidents and Evaluation Scope

The six-panel HotpotQA/2Wiki collection has zero recorded timeouts and three system events. Mistral/HotpotQA has no recorded timeouts, with 64 engine events during data collection. Mistral/MuSiQue has 2 credit-fitting, 7 correction-fitting, and 228 calibration timeouts; its final target has zero timeouts and zero system errors, with 64 engine events recorded during data collection. QuALITY/14B has 210 credit-fitting timeouts and zero downstream timeouts. Earlier incidents can affect fitted credit and calibration even when target execution is clean.

Isolated timeouts are retained under the execution protocol rather than discarded according to their score. Recovery preserves completed question–action responses; a successful response is not regen-

erated to replace an unfavorable outcome. Reported comparisons use fixed development partitions and policies. Experiments with different action spaces or generation budgets are analyzed separately.

D FULL-BANK SKILL-SELECTION DIAGNOSTIC

Figure 1a,b use nine full-action matrices. The 2Wiki/Qwen3-0.6B matrix was collected under a separately specified protocol to complete the dataset–model grid. The nine panels cover 256 MuSiQue questions and 96 questions each from HotpotQA and 2Wiki for Qwen3-0.6B, Qwen3-4B-Instruct-2507, and Qwen3-14B. Within each dataset, source IDs, question hashes, and all thirty skill meanings match across checkpoints. Banks are dataset-specific. All questions are previously inspected official-training development data, separate from ECSS's held-out comparison. The additional panel preserves the same 2Wiki questions, catalog, prompt, and budget, with its own decisions and rewards.

Each question has one recorded answer under all thirty individual skills and under no skill. MuSiQue uses 768 answer tokens: 7,936 answers per checkpoint. HotpotQA and 2Wiki use 190 answer tokens: 2,976 answers per checkpoint–dataset panel. Thus 41,664 answers enter the nine plotted panels; the separately collected SAPO triples are excluded. Temperature is 0.2 and thinking is disabled. Prompts provide the question and evidence; scoring uses the original normalized token F1 and dataset-specific alias handling. Original action/model-keyed seeds are retained.

All Agents make one temperature-zero selection call from the entire randomized catalog, with no reward, retrieval-rank, or answer-evidence input. MuSiQue uses a 320-token JSON decision/top-ten-ranking interface; only STOP or the first skill is scored. HotpotQA/2Wiki use a 96-token JSON STOP/single-choice interface. Neither uses repeated selection or answer voting. There are 1,344 selection calls in these nine panels. The historical interfaces differ from the main experiment's 128-token Agent; the panels serve as diagnostics and are analyzed by dataset.

The no-skill control replays the answer with an empty skill set. BGE Top1 uses the first entry of each question's originally frozen BGE skill ranking and replays that singleton answer. These rankings are identical across the three executors within each dataset. No answer outcomes enter the retrieval decision; neither control adds a generation call or voting.

The oracle takes the maximum recorded F1 over thirty skills and no skill for each question/checkpoint. This hindsight maximum measures observed headroom. Maximizing one draw per action amplifies generation noise, so the nominal source-question bootstrap intervals describe the recorded answers rather than achievable policy gain.

Table 6: Full-bank skill-selection diagnostic. Token F1 (↑); the gap is best observed minus Agent, with nominal source-question bootstrap 95% CIs. *4B denotes Qwen3-4B-Instruct-2507.

Dataset	Qwen	N	No skill	BGE Top1	Agent	Best obs.	Gap [95% interval]
MuSiQue	0.6B	256	8.71	9.44	9.60	18.81	9.20 [6.68,11.94]
MuSiQue	4B*	256	41.93	43.46	39.95	68.80	28.85 [23.82,34.15]
MuSiQue	14B	256	43.54	42.78	42.01	54.94	12.93 [9.73,16.36]
HotpotQA	0.6B	96	28.63	27.22	30.49	46.26	15.76 [9.99,22.31]
HotpotQA	4B*	96	72.88	69.46	66.09	84.02	17.93 [11.27,25.12]
HotpotQA	14B	96	63.54	61.72	63.94	74.83	10.89 [5.77,16.71]
2Wiki	0.6B	96	18.20	21.83	20.08	49.24	29.16 [21.09,37.27]
2Wiki	4B*	96	83.06	85.90	81.64	93.63	12.00 [6.19,18.58]
2Wiki	14B	96	65.21	64.29	67.86	79.72	11.86 [6.24,18.03]

Identity replay verifies complete coverage, matching source hashes, and no recorded answer timeouts or system errors. The frozen STOP fallbacks retain 29/96 schema placeholders on HotpotQA/0.6B and 49 placeholders plus one truncated choice on 2Wiki/0.6B. The reported small-model gap therefore combines skill selection with interface reliability.

The case in Figure 1 is a retrospective illustration of a valid, non-truncated selection failure. Displayed skill descriptions abbreviate the original catalog entries.

D.1 Cross-Backbone Credit Transfer

Figure 1c compares credit sources while holding the receiving executor, skill bank and credit estimator fixed. Each source uses the same 1,024 fitting questions and eight recorded answers per question (seven sampled skills and no skill). Direct selection uses $b + r$ for both sources, without receiver-side recalibration or semantic correction. Evaluation uses 256 previously inspected official-training questions per dataset, supplied evidence, 768 answer tokens, temperature 0.2 and the original normalized-answer F1. This cohort is separate from panels a,b. Pooled credit uses one executor's history block per fitting question, including the receiving executor, and preserves the total history count. Dashes in Table 7 mark uncollected conditions.

Table 7: Direct transfer of historical skill credit. Token F1 (↑) by receiving executor and credit source. 4B denotes Qwen3-4B-Instruct-2507; 14B, Qwen3-14B; and M7B, Mistral-7B-Instruct-v0.3.

Dataset	Executor	4B credit	14B credit	M7B credit	Pooled
MuSiQue	4B	53.49	48.81	52.17	54.14
	14B	49.31	49.22	51.31	49.32
	M7B	24.61	24.65	25.28	22.37
HotpotQA	4B	80.25	77.97	79.10	80.73
	14B	71.14	73.90	72.68	71.50
	M7B	36.60	37.02	36.43	39.42
2Wiki	4B	82.38	83.04	–	85.44
	14B	65.19	67.13	–	65.47

For the displayed 14B-to-4B direction, transferred-minus-own F1 differences are -4.68, -2.28 and $+0.66$ on MuSiQue, HotpotQA and 2Wiki; nominal paired 95% intervals are $[-8.79, -0.61]$, $[-5.60, 1.09]$ and $[-2.82, 4.11]$, respectively (10,000 question-bootstrap draws). Foreign credit also exceeds own credit on MuSiQue/14B, and pooled credit exceeds own credit on HotpotQA/M7B and 2Wiki/4B. Transfer effects are task-dependent, not uniformly harmful. The complete study uses 30,548 physical answer identities, including 12,753 exact reuses, with no new-study system errors or timeouts; historical fitting incidents remain included in the sources.

The same study also gives each source the same receiver-side calibration questions and residual-scale grid. Own and pooled direct credit nearly tie on average (58.51 versus 58.55 F1); after receiver calibration, own credit scores 60.14 versus 58.57 for pooled credit, a descriptive paired difference of $+1.57$ F1 points ($[+0.38, +2.78]$, 95% CI) across these previously inspected development panels. Own credit wins six of eight calibrated conditions, while the 2Wiki/4B pooled source remains better. Equal scored history counts do not imply equal end-to-end collection cost.

E Executor–Domain versus Shared Calibration

We isolate the outer calibration rule by holding each executor's credit, seven-feature value model, regularization, preference vectors, and support fixed. The six panels are MuSiQue, HotpotQA, and 2Wiki with 4B and 14B, each with 256 calibration and 256 target questions. All four adaptive procedures use the same six-panel calibration-label pool but group records differently: one scale globally, two executor-specific scales, three domain-specific scales, or six executor–domain scales. Each included panel has equal weight; no foreign-model rewards enter credit or value fitting. Every scale is selected from $\{0, 0.5, 1\}$ by calibration F1, with smaller scale on exact ties. Three fixed-scale controls require no outer search. In the tables, "own" denotes executor–domain calibration, and help/harm counts compare recorded model–question outcomes with Agent.

This procedure was specified after inspecting the development panels. This retrospective analysis uses previously inspected development panels; new action lists were frozen before answer-cache lookup. The 10,752 role–question evaluations reuse 2,467 exact original or completed sensitivity answers, with zero missing answers, timeouts, or system errors. Original own-calibration actions and scores replay exactly. Shared-question pairing across backbones is retained in 20,000 within-dataset bootstrap resamples; intervals condition on the existing fits, calibration and answers.

Table 8: Outer calibration controls. Token F1, exact match, and answer-level changes relative to Agent.

Scope	F1	EM	Own minus row [95% CI]	Help / harm
Executor–domain	68.16	58.79	+0.00 [+0.00, +0.00]	125 / 93
Global	67.65	58.20	+0.51 [+0.12, +0.94]	119 / 96
Executor only	67.65	58.20	+0.51 [+0.12, +0.94]	119 / 96
Domain only	67.65	58.20	+0.51 [+0.12, +0.94]	119 / 96
Fixed $\eta = 0$	67.20	57.55	+0.96 [+0.10, +1.86]	121 / 112
Fixed $\eta = 0.5$	67.39	57.94	+0.77 [+0.00, +1.57]	124 / 110
Fixed $\eta = 1$	67.65	58.20	+0.51 [+0.12, +0.94]	119 / 96

Table 9: Calibration rules by executor and domain. Token F1 and selected correction scale.

Panel	Own η	Own	Shared / fixed 1	Fixed 0	Fixed 0.5
MuSiQue / 4B	0	53.72	50.80	53.72	51.98
MuSiQue / 14B	0.5	52.54	52.40	52.54	52.54
HotpotQA / 4B	1	77.83	77.83	78.31	78.05
HotpotQA / 14B	1	74.37	74.37	71.26	72.49
2Wiki / 4B	1	82.01	82.01	81.34	82.41
2Wiki / 14B	1	68.50	68.50	66.01	66.85

Own calibration exceeds each pooled rule by $+0.509$ F1 points ($[+0.117, +0.941]$, paired-question 95% CI). The three pooled rules select identical actions and form one effective control. The difference comes from MuSiQue: $+2.915$ points on 4B and $+0.141$ on 14B, with no change in the other panels. The contrast isolates calibration grouping with credit and correction fits held fixed.

F Component and Calibration Sensitivity

These retrospective development controls use the six 4B/14B QA panels, each with 1,024 credit-fitting, 512 correction-fitting, 256 calibration, and 256 target questions. Fits and policy choices are not revised using target outcomes.

Matched-capacity and alignment controls. Direct reward regression uses the same seven features, pair-difference scaling, question weighting and ridge selection but predicts reward differences without first subtracting the credit prior. Its calibration interpolates between the prior and that prediction. Query shuffle permutes entire question-dependent credit vectors independently within fitting, calibration and target partitions, while keeping the original support. Label shuffle permutes observed value-fitting labels within each action and training fold, preserving support, missingness and action marginals; calibration labels remain real. Neither removes the original credit-fitting history. Semantic-zero instead zeros the last five fitted coefficients at inference without refitting; it is not the retrained credit-only control.

Table 4 reports the six-condition component results in the main text.

F.1 Calibration-Grid Resolution

The comparison in Table 10 changes only search resolution: nested uniform grids on the same $[0, 1]$ interval use $K = 2, 3, 5, 9, 17, 33$, or 65 points, and $K = 3$ is exactly $\{0, 0.5, 1\}$. Each grid selects its scale on the same independent calibration labels with all fits held fixed. All new actions are frozen before answer-cache lookup; all 10,752 logical target evaluations are covered by

1,840 existing answers, with no new generation. The action-union column counts distinct calibration actions, not new model calls; search time is offline CPU time.

Table 10: Calibration-grid resolution on development questions. K uniformly spaced strengths span $[0, 1]$ and are selected on independent calibration questions. Target token F1 (↑), paired differences from $K = 3$, offline search time, and distinct selected CAL question–action pairs.

Candidates	F1	Δ vs 3 [95% CI]	Search (ms)	CAL action union
2	68.14	-0.02 [-0.20, +0.11]	0.073	2309
3	68.16	+0.00 [+0.00, +0.00]	0.104	2422
5	67.81	-0.35 [-0.77, +0.04]	0.165	2498
9	67.87	-0.29 [-0.76, +0.17]	0.289	2538
17	67.67	-0.49 [-1.06, +0.06]	0.532	2551
33	67.63	-0.53 [-1.11, +0.02]	1.062	2561
65	67.67	-0.49 [-1.07, +0.07]	2.036	2563

Every nonreference macro interval in Table 10 includes zero. Denser grids increase the calibration objective without improving mean target reward. The original three-point grid remains the prespecified reference.

Measured search cost. The table measures offline CPU scale search with scores precomputed; each entry averages six panel medians from 31 randomized-order rounds. The deployed semantic-call and answer-call budget stays fixed across grids.

G Primary QA Confirmation and Repeatability Studies

The new-question confirmation, answer repeats, and calibration replicas address distinct sources of variation and are reported separately. The first two studies cover eight model–task conditions: Qwen3-4B-Instruct-2507 and Qwen3-14B on MuSiQue, HotpotQA, and 2Wiki; Mistral-7B-Instruct-v0.3 on MuSiQue and HotpotQA. They use frozen executor-specific fits, the same 30-skill-plus-STOP banks, and supplied-context answers at 768 tokens and temperature 0.2. Identical question–executor–action answers are reused across selector roles.

G.1 New-Question Confirmation

After excluding 16,644 previously used IDs and 18,878 normalized-question hashes, we draw 256 calibration and 256 target questions per dataset from official training splits. Questions may share articles or entities. Historical fits, prompts, search grids, and tie rules remain fixed, while the original calibration procedure uses the new calibration outcomes. Each question receives a full-bank probability call and a direct JSON choice; target actions precede target answers. Collection comprises 36,068 physical calls, including 8,192 selector calls, with no timeouts or system errors. Independent replay reproduces actions, scores, and calibration choices from the raw responses.

ECSS (credit only) uses $b + \rho r$; ECSS (score fusion) adds semantic preference. The supplementary *preference argmax* takes the top probability action, while Agent makes a direct JSON choice.

Uncertainty and paired changes. Paired 95% CIs use 20,000 within-dataset question-bootstrap draws, shared across backbones. The prespecified contrast is ECSS (full) minus Agent; other contrasts are descriptive. Table 14 counts observed answer-level changes; equal F1 can arise from different actions. The contrasts describe recorded answers under fixed policies.

Table 11: New-question confirmation. Token F1 by executor and dataset.

Panel	Agent	Preference argmax	BGE	Top3	STOP	ECSS (credit only)	ECSS (score fusion)	ECSS (full)
MuSiQue / 4B	51.72	53.19	53.95	44.90	53.45	54.49	51.90	49.44
MuSiQue / 14B	49.72	49.03	50.35	49.98	48.49	50.04	49.22	49.68
MuSiQue / Mistral7B	22.16	23.18	21.39	19.82	21.18	26.77	26.77	27.26
HotpotQA / 4B	72.78	75.10	79.74	73.64	73.74	81.47	81.84	79.89
HotpotQA / 14B	72.88	73.74	70.74	71.02	70.51	75.36	75.36	74.35
HotpotQA / Mistral7B	35.96	36.56	36.52	25.42	38.14	44.46	44.46	44.21
2Wiki / 4B	83.75	84.01	82.47	72.05	78.04	82.36	82.68	82.77
2Wiki / 14B	67.31	67.91	66.97	67.85	66.00	66.20	68.12	68.58
Equal-panel mean	57.04	57.84	57.77	53.08	56.20	60.14	60.04	59.52

Table 12: Question-bootstrap SD for the new-question results. Token F1 points by condition and equal-panel average.

Condition	No skill	BGE Top1	SAPO Top3	Agent	ECSS (credit only)	ECSS (full)
MuSiQue / 4B	2.88	2.86	2.89	2.89	2.83	2.83
MuSiQue / 14B	2.74	2.74	2.77	2.71	2.73	2.73
MuSiQue / M7B	2.02	1.97	1.82	1.95	2.24	2.31
HotpotQA / 4B	2.64	2.37	2.64	2.66	2.26	2.39
HotpotQA / 14B	2.59	2.56	2.54	2.52	2.48	2.53
HotpotQA / M7B	2.48	2.44	1.88	2.52	2.71	2.71
2Wiki / 4B	2.47	2.22	2.69	2.15	2.21	2.17
2Wiki / 14B	2.60	2.60	2.59	2.62	2.62	2.55
Equal-panel Avg.	1.13	1.12	1.07	1.12	1.14	1.16

Table 13: New-question confirmation. Exact match by executor and dataset.

Panel	Agent	Preference argmax	BGE	Top3	STOP	ECSS (credit only)	ECSS (score fusion)	ECSS (full)
MuSiQue / 4B	43.75	44.14	44.53	37.89	44.14	44.53	42.19	39.45
MuSiQue / 14B	37.11	37.89	38.67	38.67	37.11	37.89	37.11	37.50
MuSiQue / Mistral7B	10.16	12.11	10.16	8.20	9.38	14.45	14.45	15.62
HotpotQA / 4B	67.97	70.70	74.61	69.92	68.75	76.17	76.56	75.78
HotpotQA / 14B	64.45	65.23	62.50	62.11	62.11	68.75	68.75	67.97
HotpotQA / Mistral7B	23.44	22.66	21.88	9.38	23.05	32.81	32.81	32.81
2Wiki / 4B	78.52	78.12	76.95	68.75	73.05	76.56	76.95	76.56
2Wiki / 14B	55.86	55.47	55.47	56.64	53.52	54.30	55.86	56.25
Equal-panel mean	47.66	48.29	48.10	43.95	46.39	50.68	50.59	50.24

Table 14: **Answer-level effects of ECSS (full).** Paired token F1 outcomes relative to Agent and ECSS (credit only).

Reference selector	Improved	Worsened	Tied
Agent	269	202	1,577
ECSS (credit only)	64	73	1,911

G.2 Fixed-Policy Answer Repeats

This earlier, separate cohort uses 128 previously inspected questions per condition and two fresh answer streams, with histories, fits, calibration, and Agent choices fixed. It comprises 12,168 physical answers and 16,384 role evaluations, with no timeout and one recovered connection error. Table 16 reports sample SD across the two streams; Table 15 reports paired contrasts. This study measures answer-generation variation with histories, fits, and policies fixed.

Table 15: Paired token-F1 contrasts in two separate QA cohorts. Each entry reports a mean difference [paired question-bootstrap 95% CI].

ECSS minus	New questions	Fixed-policy answer repeats
Agent	+2.49 [+1.14, +3.85]	+1.31 [-0.30, +2.96]
Preference argmax	+1.68 [+0.47, +2.91]	+2.10 [+0.51, +3.69]
BGE Top1	+1.76 [+0.47, +3.06]	+2.74 [+1.18, +4.31]
SAPO Top3	+6.44 [+4.95, +7.94]	+6.41 [+4.58, +8.25]
No skill	+3.33 [+1.95, +4.72]	+3.36 [+1.62, +5.10]
ECSS (credit only)	-0.62 [-1.35, +0.11]	-0.08 [-1.05, +0.94]
ECSS (score fusion)	-0.52 [-1.28, +0.25]	+0.62 [-0.66, +1.92]

Table 16: Answer-generation repeatability. Equal-panel token F1, mean ± sample SD across two answer streams.

Method	F1 ↑ (mean ± SD)
No skill	57.86 ± 0.14
BGE Top1	58.48 ± 0.15
SAPO Top3	54.82 ± 0.02
Agent	59.91 ± 0.54
Preference argmax	59.13 ± 0.35
ECSS (credit only)	**61.30** ± 0.16
ECSS (score fusion)	60.60 ± 0.27
ECSS (full)	61.22 ± 0.74

G.3 Independent Calibration Replicas

The MuSiQue/4B and MuSiQue/14B follow-up varies calibration evidence while holding the original 1,024-question credit fit and 512-question correction fit fixed for each executor. Three disjoint 256-question calibration sets select parameters independently; all use the same 256 target questions and freeze actions before target answers. The official-training allocation contains 1,024 distinct questions. Model, bank, prompt, evaluator, and 768-token budget are unchanged. All 13,413 calls completed without errors or timeouts, and raw answers and role scores were replayed.

Table 17 reports the unselected mean and sample SD across the three calibration sets. Because target questions and question–action answers are shared, the daggered controls have zero SD by construction. The remaining SDs measure sensitivity to calibration evidence with fits and answers fixed.

Table 17: **Calibration-set repeatability.** Target token F1, mean ± sample SD across three calibration sets.

Method	MuSiQue / 4B	MuSiQue / 14B
No skill†	**50.94** ± 0.00	47.64 ± 0.00
BGE Top1†	48.57 ± 0.00	53.87 ± 0.00
SAPO Top3†	46.28 ± 0.00	52.82 ± 0.00
Agent†	50.08 ± 0.00	51.32 ± 0.00
Preference argmax†	50.05 ± 0.00	52.64 ± 0.00
Credit prior†	47.55 ± 0.00	53.18 ± 0.00
ECSS (credit only)	49.38 ± 2.83	**54.12** ± 0.81
ECSS (score fusion)	49.38 ± 2.83	52.95 ± 0.40
Credit-only correction	47.78 ± 0.59	53.18 ± 0.00
Shuffled-semantic correction	47.56 ± 0.00	53.18 ± 0.00
ECSS (full)	47.43 ± 0.11	53.40 ± 0.19

The frozen action plans help explain ECSS (full)'s lower calibration-set SD. The minimum pairwise target-action agreement across replicas is 221/256 on 4B and 238/256 on 14B for ECSS (full), versus 36/256 and 183/256 for ECSS (credit only), and 36/256 and 18/256 for ECSS (score fusion). ECSS (full) holds historical credit fixed while scaling its learned correction; the controls can instead rescale credit. Since answers are shared across replicas, changed actions account for the observed score variation. On 4B, ECSS (score fusion) always assigns zero weight to preference and exactly matches ECSS (credit only).

ECSS (full) has smaller sample SD than ECSS (credit only) on both executors (0.11 versus 2.83 on 4B; 0.19 versus 0.81 on 14B), but a lower mean. The paired ECSS–credit difference is -1.34 F1 points ($[-2.16, -0.59]$, 95% CI); the corresponding differences are -0.28 versus Agent ($[-2.95, +2.40]$) and $+0.87$ versus Top3 ($[-2.45, +4.21]$). These shared-question intervals and calibration-set SDs describe different quantities. This MuSiQue follow-up has its own fixed-fit evidence base.

H Separate Mistral/2Wiki Executor Extension

We evaluate the unchanged seven-feature ECSS on an additional Mistral-7B-Instruct-v0.3/2Wiki panel. Its 2,048 distinct official-training questions comprise 1,024 credit-fitting, 512 correction-fitting, 256 calibration, and 256 target questions. Rewards, fits, and calibration are newly obtained for this executor under the 768-token answer protocol. We report it as a separate executor extension.

Table 18: Mistral/2Wiki extension. Token F1 and exact match (↑).

Method	F1 ↑	EM ↑
No skill	27.43	11.72
BGE Top1	28.92	13.28
SAPO Top3	24.01	6.64
Agent	25.56	10.16
Preference argmax	25.82	9.77
Credit prior	28.96	12.11
ECSS (credit only)	30.80	14.84
ECSS (score fusion)	30.80	14.84
Credit-only correction	**30.90**	**15.62**
Shuffled-semantic correction	30.89	13.67
ECSS (full)	29.08	11.72

Collection used 14,740 deduplicated model calls (8,192 credit fitting, 2,886 correction fitting, 1,413 calibration, and 2,249 target), with no system errors or timeouts. One synthetic interface probe was outside the evaluation. Independent replay reproduces responses, actions, scores, calibration, and paired analysis; identical question–action answers are shared across roles without voting.

The nominal 20,000-draw paired-question bootstrap gives the following F1-point differences (95% intervals): $ECSS$ − Top3 $= +5.065$ ($[+1.127, +9.089]$), $ECSS$ − Agent $= +3.521$ ($[-0.103, +7.173]$), and $ECSS$ − pure credit $= -1.721$ ($[-5.095, +1.600]$). Calibration chooses full correction for all learned correction models. ECSS (credit only) and ECSS (score fusion) select skill `2wiki_native_22` on all 256 targets, yielding a fixed policy.

ECSS changes the historical choice on 149 questions: 35 answers improve, 38 worsen, and 76 tie. Gains of 5.141 and losses of 6.862 F1 points yield the -1.721 net difference. Historical credit is useful on this panel, while semantic correction adds no observed benefit.

I GSM8K: Executor-Specific Arithmetic Extension

The arithmetic extension uses a new bank of thirty fixed arithmetic procedures, with no QA credit or correction weights transferred. Each of Qwen3-4B-Instruct-2507 and Mistral-7B-Instruct-v0.3 uses 1,024 credit-fitting, 512 correction-fitting, 256 calibration, and 256 target questions. The 2,048

distinct source questions come from official GSM8K training data and are shared across backbones. Gold worked solutions are hidden from the answer model.

The scorer follows the upstream delimiter-based extraction and normalized numeric-string comparison. Table 19 reports upstream answer accuracy. The answer budget remains 768 tokens, temperature 0.2, one generation, no tools or answer voting. Five Agent selections on 4B require the declared format normalization; none do on Mistral. All twelve selector roles are retained. Collection completes 14,551 and 14,734 unique calls, respectively, with zero system errors and timeouts. The exported scores replay all 256 binary outcomes per selector and match the archived raw answers.

Table 19: **GSM8K comparisons and component ablations.** Answer accuracy (↑).

Method	Qwen3-4B	Mistral-7B
No skill	94.92	32.03
BGE Top1	94.92	20.70
SAPO Top3	92.97	25.39
Agent	94.92	23.05
Preference argmax	94.14	23.05
Credit prior	**95.31**	40.62
ECSS (credit only)	94.92	41.02
ECSS (score fusion)	**95.31**	41.02
Credit-only correction	**95.31**	**42.58**
Shuffled-semantic correction	**95.31**	41.41
ECSS (full)	**95.31**	41.41

Independent calibration selects $\eta = 0.5$ on 4B and $\eta = 1$ on Mistral. For 4B and Mistral, respectively, the ECSS–Agent accuracy differences are $+0.39$ and $+18.36$ points, with paired-question 95% intervals $[-1.95, +2.73]$ and $[+11.33, +25.39]$. Against Top3 the corresponding differences are $+2.34$ and $+16.02$ points, with intervals $[-0.39, +5.08]$ and $[+8.59, +23.44]$. These intervals condition on the recorded answer stream. On both executors the ECSS–credit difference is $+0.39$ points; on Mistral, historical selection accounts for nearly all of the gain over Agent.

J HISTORY-BUDGET SENSITIVITY

This prespecified supplement uses MuSiQue/4B and /14B with the first calibration replica and its shared 256 target questions. Candidate policies precede calibration outcomes, and final actions precede target outcomes. Nested comparisons reuse the recorded answer pool to isolate the effect of credit-fitting history size.

Nested credit-fitting budgets. Table 20 uses nested histories of 128, 256, 512 and 1,024 questions, with real and shuffled-label fits. Each independently selects $\rho \in \{0, 0.5, 1\}$ on the same first calibration replica. The logical credit-collection cost is $8N$ answers; the reference also uses 512 correction-fitting questions. On 4B all settings make the same target choices. On 14B the full real history exceeds its shuffled control, while intermediate budgets are nonmonotonic.

Table 20: Historical credit data size. Target token F1 (↑).

Credit-FIT questions	4B real	4B shuffled	14B real	14B shuffled
128	52.65	52.65	51.96	52.95
256	52.65	52.65	51.38	50.44
512	52.65	52.65	52.60	52.60
1024	52.65	52.65	54.58	50.75