CAMS: RESIDUAL CAPABILITY ALLOCATION FOR CONTINUED TOOL-USE SFT

Anonymous authors
Paper under double-blind review

ABSTRACT

Data selection for tool-use supervised fine-tuning commonly ranks records by global quality, relevance, or estimated utility. Staged adaptation raises a more specific question: once a model has completed initial adaptation, which remaining training records should form the next batch for continued training under a fixed budget? We define this setting as marginal continuation selection and introduce Capability-Aware Marginal Selection (CAMS). CAMS matches what the current model still needs with what supervision each candidate record provides. A frozen train-side probe supplies capability accuracies C_k, yielding $D_k = 1 - C_k$ as a residual-headroom proxy; gold tool calls and TRAIN statistics supply observable supervision demands V_k. CAMS ranks candidates by $S(\tau \mid \theta) = \sum_k D_k(\theta)V_k(\tau)$ and admits intact records under the continuation budget. The current instantiation uses two auditable demand axes: multi-binding structure and API exposure scarcity. We evaluate on the frozen API-Bank train-side confirmation set (FCS): 256 converted API-call records disjoint from training, the capability probe, and selector-construction records. On Qwen3-8B under the frozen API-Bank continuation protocol, CAMS achieves 35.29% exact API-call correctness on FCS, the highest mean among five baselines fixed before confirmation evaluation. It exceeds the strongest pre-frozen baseline by 1.83 percentage points and same-state Random by 2.65 points, with positive CAMS–Random differences in all three training states. Post-hoc descriptive CAMS–Random differences are 10.0 points on multi-argument calls and 13.43 points on rare-API calls. CAMS frames continued tool-use SFT as model-state-aware residual capability allocation: matching current needs to observable supervision in the remaining data, rather than repeating a static global ranking.

1 INTRODUCTION

Static example ranking summarizes global quality, relevance, or general utility. After initial tool-use supervised fine-tuning (SFT), the model has acquired some behaviors while retaining a non-uniform residual capability profile. Continued SFT therefore poses a model-state-aware allocation problem, not merely another round of static ranking: which available supervision matches what this model still needs?

We study *marginal continuation selection*: choosing records for continued training from the same adapted state under a limited budget. We frame this problem as *residual capability allocation*. A frozen train-side probe measures model-side residual need, while gold calls and TRAIN statistics describe candidate-side observable supervision demand. The allocation question is what the model still needs and what supervision the candidate supplies. CAMS makes this compatibility explicit through a first-order capability-factorized surrogate. Here "first-order" means retaining additive individual capability terms while omitting pairwise and higher-order interactions; it is not a Taylor approximation of training dynamics.

We introduce *Capability-Aware Marginal Selection* (CAMS). Its score $S(\tau \mid \theta_A) = \sum_k D_k(\theta_A)V_k(\tau)$ combines a residual-headroom proxy $D_k = 1 - C_k$ with an observable demand indicator V_k. The current tool-use instantiation represents multi-binding structure through calls with at least two arguments, and exposure scarcity through APIs in the bottom half of TRAIN-frequency identity ranks. These two auditable demand axes define what supervision a record supplies; the

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.

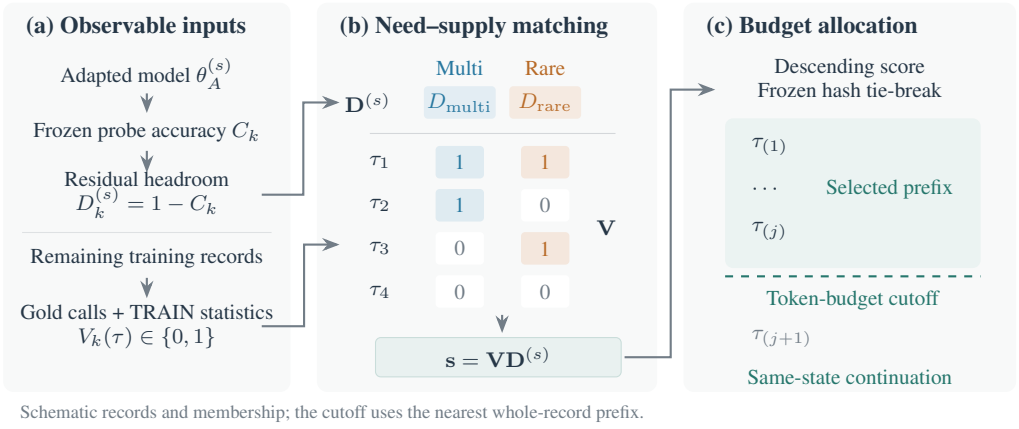
deficits determine how much the current model needs it. Capability definition and prioritization are distinct.

CAMS allocates the continuation budget by deterministic score ranking, a frozen hash tie-break, and a serialized-token cutoff. On the independently frozen FCS surface, it reaches 35.29% exact correctness, achieving the highest FCS mean among five pre-frozen baselines, exceeding our frozen embedding-relevance baseline TIS by 1.83 percentage points and same-state Random by 2.65 points. All three state-specific CAMS–Random contrasts are positive.

Construct and operational audits clarify what this allocation achieves. The selected records have higher residual-capability coverage per record, in total, and per serialized token than the audited frozen baselines. The both-demand pool is sufficient to absorb the entire budget, so the three states select identical members. The explicit decomposition makes this saturation regime directly diagnosable: state-specific deficits change scores but not membership.

Our contributions are:

1. **Marginal continuation selection.** We formulate continued SFT as model-state-aware allocation of a limited budget against the residual capability profile after initial adaptation.
2. **Need–supply matching.** We introduce CAMS, a simple, parameter-free, auditable residual-capability coverage surrogate combining model-side residual headroom with candidate-side observable supervision demands.
3. **Controlled continuation evidence.** CAMS achieves the highest FCS mean among five pre-frozen baselines, while improving over same-state Random with positive direction in all three FCS states.
4. **Auditable allocation behavior.** We audit the demand definitions, deficit interpretation, and actual selected coverage, and identify when candidate-pool saturation makes state weighting inactive for membership.



Schematic records and membership; the cutoff uses the nearest whole-record prefix.

Figure 1: Residual capability allocation after initial adaptation. The frozen probe supplies residual need, gold calls and TRAIN statistics supply observable demands, and deterministic matching allocates the continuation budget.

2 Related Work

Tool-use agents and function-calling benchmarks. Toolformer studies learning when to call external tools (Schick et al., 2023); ToolLLM and ToolACE study tool-use training data at scale (Qin et al., 2024; Liu et al., 2024b). Gorilla, BFCL and API-Bank evaluate API-use behavior (Patil et al., 2023; 2025; Li et al., 2023), while τ-bench and AppWorld include interactive state and execution (Yao et al., 2024; Trivedi et al., 2024). Our empirical unit is a converted API-Bank context–call record. We study selection of these intact records for staged SFT.

Data selection for language-model training. DSIR uses importance resampling for distribution-oriented selection (Xie et al., 2023); AlpaGasus and Deita explore instruction-data quality and selection (Chen et al., 2023; Liu et al., 2024a). Model-dependent utility is also established: DELIFT defines pairwise utility relative to current model capabilities and supports instruction, task-specific, and continual fine-tuning (Agarwal et al., 2024). Recent continual or recovery settings likewise use dynamic data selection based on acquired knowledge or capability degradation (Maharana et al., 2025; He et al., 2026). CAMS instantiates model-dependent selection through interpretable binary tool-call demands and probe-derived deficits in a fixed tool-use continuation protocol.

Influence and gradient-based selection. Influence functions and TracIn estimate training-example effects through model derivatives or checkpoints (Koh & Liang, 2017; Pruthi et al., 2020). LESS uses optimizer-aware low-rank gradient features to select instructions for target capabilities (Xia et al., 2024). CAMS instead uses explicit candidate-observable demand indicators and probe-derived residual deficits without computing selection gradients. In our comparisons, LESS denotes our frozen local implementation of the LESS selection idea; it is not a full reproduction of the original LESS protocol (Appendix B).

Capability targeting and robust selection. Multi-environment learning studies how performance can fail when average risk hides group or domain variation. Invariant risk minimization, group distributionally robust optimization, and DomainBed provide related language and caution for environment-dependent generalization (Arjovsky et al., 2019; Sagawa et al., 2020; Gulrajani & Lopez-Paz, 2021). Multi-task optimization also studies gradient conflict and agreement across objectives (Chen et al., 2018; Yu et al., 2020). CAMS shares the concern that aggregate performance can hide heterogeneous capability profiles and supervision needs, and addresses them through pre-continuation data selection: the capability slices are used to rank candidate records before SFT, and the final claim is judged by downstream confirmation performance rather than by the slice objective alone.

Diversity and coverage selection. Subset selection has a long history in relevance-diversity trade-offs, including maximal marginal relevance, submodular facility-location objectives, and determinantal point processes (Carbonell & Goldstein, 1998; Krause et al., 2008; Kulesza & Taskar, 2012). CAMS measures residual-capability coverage through its explicit need–supply surrogate. A coverage selector can spread examples across tools without knowing which tool-use capabilities are weak in the current model. CAMS instead targets two pre-specified demands, multi-argument calls and rare APIs, and treats diversity statistics as diagnostics rather than success criteria.

External benchmark substrate alignment. Interactive tool-use benchmarks differ not only in tasks, but also in serialization, tool schemas, user simulators, state transitions, and evaluators. This matters for data-selection claims. Our external discussion distinguishes direct adapter transfer, which assesses weight portability, from within-substrate data selection, which assesses selector portability. A clean external selector test should retrain Random, TIS, LESS, and CAMS inside the new benchmark's own train substrate, verify healthy Random adaptation there, and then evaluate on a held-out surface without changing the selector after outcomes are observed.

3 Problem Formulation

Let $\mathcal{D} = \{\tau_i\}_{i=1}^{N}$ be a frozen pool of converted API-call records. Each contains a context, tool metadata, and a gold assistant call. We select records as indivisible units. The serializer masks context tokens and admits the supervised answer under a fixed length limit (Appendix B).

Stage 1 trains a state $\theta_A^{(s)}$ from a common subset A, with training state $s \in \{0, 1, 2\}$. Stage 2 selects $\mathcal{B}^{(s)} \subseteq \mathcal{D} \setminus A$ under a serialized-token target b_2 and continues from that state:

$$\sum_{\tau \in \mathcal{B}^{(s)}} \text{tokens}_{\text{serialized}}(\tau) \approx b_2. \tag{1}$$

All methods share branchpoints, serialization, optimizer and scheduler semantics, and the same confirmation protocol. They need not have identical supervised-token counts or optimization steps.

The approximation follows a frozen nearest-prefix rule rather than splitting records. Here "marginal" refers to continuation after Stage 1.

CAMS uses two binary labels: $V_{\mathrm{multi}}(\tau) = 1$ when a supervised call has at least two arguments, and $V_{\mathrm{rare}}(\tau) = 1$ when a call uses a bottom-half train-frequency API. A frozen train-side probe supplies capability accuracies $C_k(\theta_A^{(s)})$. The method below combines these accuracies with the labels without using held-out confirmation outcomes for selection.

4 CAMS: Residual Capability Allocation

4.1 Observable supervision demands

CAMS assumes a small pre-specified set of auditable demand axes and matches them to an adapted model's residual profile. Each binary indicator $V_k(\tau)$ identifies whether a record supplies the corresponding supervision, while the probe-derived deficit determines its current priority. Demand definition and capability prioritization are distinct. The indicators are observable before continuation and computed deterministically without a learned supervision-strength model. Our current instantiation uses two axes: multi-binding structure and API exposure scarcity.

Multi-binding structure. We set $V_{\mathrm{multi}}(\tau) = \mathbf{1}[n(\tau) \geq 2]$. The historical parser takes the maximum number of keyword bindings among valid gold calls in the first supervised assistant message. Two is the minimal deterministic boundary at which multiple argument bindings must be coordinated. Appendix D evaluates the support and accuracy profile of this structural demand.

Exposure scarcity. We set $V_{\mathrm{rare}} = 1$ when a record's API identities intersect the frozen rare inventory. TRAIN frequency counts each distinct API at most once per record across 6,839 valid converted TRAIN records. Sorting 1,531 identities by ascending frequency and then API name, the first 765 form the rare set. These identities cover 11.95% of TRAIN records and 11.31% of eligible Stage-2 candidates. The split is a frozen, tuning-free median-rank discretization of TRAIN-library exposure scarcity. For interpretation, let $p_{\mathrm{TRAIN}}(a)$ be an API's frequency divided by total API frequency mass. Lower-frequency identities have higher surprisal $I(a) = -\log p_{\mathrm{TRAIN}}(a)$, making rare demand a coarse binary proxy for statistical exposure scarcity. Surprisal describes the demand axis; selection uses the residual-capability coverage score below.

The selection audit supports these axes through semantic meaning, pre-continuation observability, deterministic actionability, and sufficient support. These axes provide two concrete, auditable instantiations of the interface. Appendix D.2 documents other candidate axes and the provenance of historical Capability Response.

4.2 Model-side residual need

A frozen train-side probe supplies capability accuracy $C_k(\theta_A^{(s)})$ for each Stage-1 state. We define

$$D_k^{(s)} = 1 - C_k(\theta_A^{(s)}) \tag{2}$$

as a simple, parameter-free proxy for residual headroom. Because $C_k \in [0, 1]$, the deficit is bounded in $[0, 1]$ and decreases monotonically as measured accuracy increases. It directly represents residual empirical error, or remaining headroom under the probe metric. Confirmation outcomes do not enter D_k. A descriptive calibration audit is reported in Appendix E.

4.3 From need–supply matching to the CAMS score

CAMS uses a simple, parameter-free, auditable need–supply surrogate, factorized by capability, to allocate a limited continuation budget relative to the current model state. For candidate τ and the adapted state $\theta_A^{(s)}$, it computes

$$S_{\mathrm{CAMS}}(\tau \mid \theta_A^{(s)}) = \sum_k D_k^{(s)} V_k(\tau). \tag{3}$$

Design rationale. For capability k, D_k describes model-side residual need and V_k describes candidate-side observable supervision demand. A record receives credit only when the model retains measured need and the record supplies the corresponding supervision. Multiplication $D_k V_k$ enforces this compatibility: zero need or absent demand contributes zero. Summation credits records that cover multiple residual needs. The result measures residual-capability coverage.

The dot product provides a simple parameter-free separable score with four transparent properties. *Residual-need monotonicity* means that increasing D_k cannot decrease the score for fixed nonnegative demands. *Demand gating* gives absent demands zero contribution. *Additive multi-capability coverage* gives each supplied demand its own contribution. *Parameter-free auditability* makes every contribution recoverable from the frozen probe accuracy and demand indicator. Together, these properties make the selection rationale inspectable term by term.

The measured deficits are the only weights in this scoring layer. This parameter-free construction preserves inspectable semantics and keeps selection independent of confirmation-tuned coefficients. Richer capability interactions remain future work.

Record demand	Multi contribution		Rare contribution		Total score
Both	D_{multi}	+	D_{rare}	=	$D_{\text{multi}} + D_{\text{rare}}$
Multi only	D_{multi}	+	0	=	D_{multi}
Rare only	0	+	D_{rare}	=	D_{rare}
Neither	0	+	0	=	0

Figure 2: Demand gating and additive residual-capability coverage. Each supplied demand contributes its measured deficit; an absent demand contributes zero. The four rows enumerate the symbolic scores for each demand combination.

4.4 Deterministic budget allocation

Candidates are ranked by descending score with an ascending frozen hash tie-break. At the first crossing of the serialized-token target, the implementation chooses the nearer of the immediately preceding and following prefixes, retaining whole records. A separate frozen hash determines training order. CAMS requires no learned selector weights, selection gradients, or diversity coefficient; probe evaluation still requires inference, and end-to-end selection cost has not been measured.

Algorithm 1 CAMS: residual capability allocation

Require: Candidate records $\mathcal{D}$, core A, state $\theta_A^{(s)}$, frozen probe P, demands $\{V_k\}$, token target b_2

for each capability k **do**
  Measure $C_k(\theta_A^{(s)})$ on P; set $D_k^{(s)} = 1 - C_k$
end for
for each $\tau \in \mathcal{D} \setminus A$ **do**
  Compute $V_k(\tau)$ from gold calls; score $S(\tau) = \sum_k D_k^{(s)} V_k(\tau)$
end for
Rank by descending S and ascending frozen hash
Select the nearer prefix before/after crossing b_2
return Continuation subset, ordered by the frozen training hash

5 CONTROLLED CONTINUATION EVIDENCE ON FCS

5.1 PROTOCOL AND COMPARATOR

The experiment uses API-Bank train-side partitions and Qwen3-8B in non-thinking mode. Stage 1 trains a common core A in three states. Each selector continues from the corresponding $\theta_A^{(s)}$ using BF16 LoRA with rank 8, alpha 16, zero dropout, AdamW at 2×10^{-4}, maximum length 2,048, and one pass through admitted records. The Stage-2 target is 33,118 serialized tokens, with 65,703 targeted across both stages. Matched admission budgets do not imply matched supervised exposure or optimizer steps.

FCS contains 256 converted API-call records and is disjoint from training, the capability probe, and selector-construction records. It is a frozen API-Bank train-side confirmation set, not an official hidden benchmark test. The metric is exact API-call correctness, not interactive task completion.

Random, TIS, LESS, StaticAgreement, and SCMA were frozen before confirmation evaluation, including selected subsets and checkpoints. TIS is our frozen embedding-relevance baseline (Appendix B). Base and A-only are reference conditions. Random has three subset proposals crossed with three training states; the paired comparator averages proposals within each state. Other selectors have three states. LESS denotes our frozen local implementation of the LESS selection idea; it is not a full reproduction of the original LESS protocol (Appendix B). Later mechanism controls remain separate from these five baselines.

5.2 ALLOCATING USEFUL CONTINUATION HEADROOM

A-only reaches 0.2422 on FCS, while same-state Random continuation reaches 0.3264, an 8.42-point increase. Thus useful headroom remains after initial adaptation. CAMS asks how to allocate that continuation budget using the measured residual profile.

Table 1: FCS exact API-call correctness. CAMS achieves the highest FCS mean exact correctness among five pre-frozen baselines.

Condition	FCS mean $\pm$ model SD
Base	0.0625
A-only	0.2422 ± 0.0195
Random	0.3264 ± 0.0162
TIS	0.3346 ± 0.0081
LESS	0.3294 ± 0.0200
StaticAgreement	0.3294 ± 0.0098
SCMA	0.3047 ± 0.0141
CAMS	$\mathbf{0.3529} \pm 0.0126$

Note: SD is across nine Random models or three models for other adapted conditions; Base is one model. These SDs are not confidence intervals for paired gains.

CAMS improves over TIS by 1.83 percentage points and over same-state Random by 2.65 points, with positive CAMS–Random differences in all three states. States share evaluation tasks, so this direction pattern does not constitute three independent task replications. We summarize the comparison through the mean difference and state-specific directions.

5.3 DESCRIPTIVE TARGET-SLICE OUTCOMES

On FCS, CAMS–Random gains are 10.00 points for multi-argument exactness and 13.43 points for rare-API exactness, compared with 2.65 points overall. These post-hoc outcome slices localize the observed improvements. The appendix reports their underlying means; demand definitions, deficits, and subsets were frozen before these outcomes were observed.

6 Analysis and Discussion

6.1 Residual-capability coverage

A train-side audit recomputes the historical score on all 2,016 eligible candidates and verifies the 24 frozen method–state subsets against their inventories. CAMS has higher mean residual-capability coverage, total coverage, and coverage per 1,000 serialized tokens than every audited frozen baseline in all three states. Relative to the candidate pool, its multi, rare, and both-demand record enrichment ratios are $4.157\times$, $8.842\times$, and $16.525\times$; all 32 selected records have both demands (Appendix F). CAMS operationally concentrates continuation budget on records covering its measured residual needs.

6.2 Regime diagnosis under candidate-pool saturation

The score decomposition identifies when state weights affect membership. All measured deficits are positive in the tested API-Bank states. A both-demand record therefore strictly outranks either single-demand category:

$$D_{\text{multi}} + D_{\text{rare}} > \max(D_{\text{multi}}, D_{\text{rare}}).$$

The both-demand pool contains 122 records and 134,121 serialized tokens, or 4.05 times the 33,118-token Stage-2 target. All three states order categories as both, multi-only, rare-only, then neither. Within that category every record has the same score for a given state, so the common hash tie-break and cutoff select the same 32 records (33,027 tokens and 32 APIs) in all three states. Operationally, allocation takes a frozen hash-ordered token prefix from the both-demand pool. State-specific deficit magnitudes affect score values but not subset membership in this saturated regime.

The explicit score decomposition thus makes the saturation regime directly diagnosable.

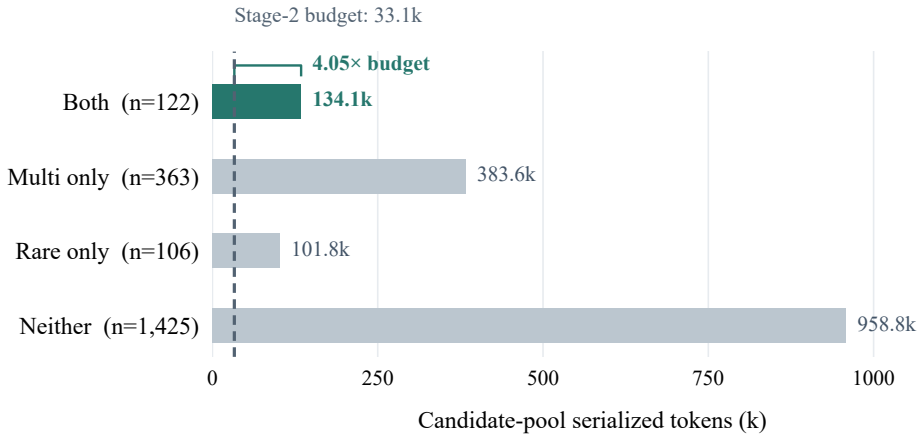


Figure 3: Both-demand supply exceeds the Stage-2 budget. The frozen tie-break and budget cutoff select the same prefix in all three states.

6.3 Demand composition and mechanism controls

The frozen comparison reports the observed CAMS–Random difference under the original protocol. Later controls on reused FCS assess mechanism attribution: coverage-matched and demand-shuffled controls can exceed CAMS in this saturated regime despite very different subset compositions (Appendix G.1). The target-slice results describe where gains occur; the causal contributions of demand labels and deficit weights remain unresolved.

6.4 Record utility depends on the training intervention

The observed advantage depends on the continuation intervention: supervised-exposure matching does not preserve it, and fixed-order smaller-budget results vary in direction. Appendix G.2 reports the exposure and exact-label/update studies, and Appendix G.3 reports budget sensitivity; these post-hoc diagnostics reuse FCS.

Scope. The findings concern continued adaptation on static API-call records under the specified admission and training protocol. Appendix H reports negative AppWorld weight-transfer and custom STATE-derived selector-portability results, which delimit extrapolation to interactive tasks and other selection settings.

7 Conclusion

We formulated continued SFT as model-state-aware residual capability allocation, rather than another static global ranking. CAMS provides a simple, parameter-free, auditable rule matching bounded residual headroom to observable supervision in the remaining data, with deterministic budget allocation. On FCS it reaches 35.29% exact correctness, the highest FCS mean among five pre-frozen baselines, and improves over same-state Random by 2.65 points with positive differences in all three states.

Construct audits support interpreting the current axes as multi-binding structure and exposure scarcity. Together, the findings support an auditable allocation rule for the tested continuation setting, linking residual capability needs to observable supervision and exposing when saturation makes membership insensitive to state weights.

AI Use Disclosure

Generative AI tools were used to assist with research workflow management, code implementation, result-check organization, figure/table drafting, literature-search support, reference formatting, and manuscript drafting/editing. They were also used to help identify claim-evidence mismatches and formatting issues. The experiments, numerical claims, and final wording were checked against repository artifacts by the authors, and the authors take responsibility for the final content, including any text, claims, code, or artifacts produced with AI assistance.

References

Ishika Agarwal, Krishnateja Killamsetty, Lucian Popa, and Marina Danilevksy. Delift: Data efficient language model instruction fine tuning, 2024. URL https://arxiv.org/abs/2411.04425.

Martin Arjovsky, Leon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization, 2019. URL https://arxiv.org/abs/1907.02893.

Jaime Carbonell and Jade Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1998. URL https://dl.acm.org/doi/10.1145/290941.291025.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, and Hongxia Jin. Alpagasus: Training a better alpaca with fewer data, 2023. URL https://arxiv.org/abs/2307.08701.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International Conference on Machine Learning*, 2018. URL https://proceedings.mlr.press/v80/chen18a.html.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=lQdXeXDoWtI.

Bowei He, Lihao Yin, Huiling Zhen, Xiaokun Zhang, Mingxuan Yuan, and Chen Ma. Paser: Post-training data selection for efficient pruned large language model recovery. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/180d2acb13633fe78688d0d2347c731f-Abstract-Conference.html.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning*, 2017. URL https://proceedings.mlr.press/v70/koh17a.html.

Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. In *Journal of Machine Learning Research*, 2008. URL https://www.jmlr.org/papers/v9/krause08a.html.

Alex Kulesza and Ben Taskar. *Determinantal Point Processes for Machine Learning*. Foundations and Trends in Machine Learning, 2012. URL https://arxiv.org/abs/1207.6083.

Minghao Li, Feifan Song, Bowen Yu, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li. Api-bank: A comprehensive benchmark for tool-augmented large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. URL https://aclanthology.org/2023.emnlp-main.187/.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. In *International Conference on Learning Representations*, 2024a. URL https://openreview.net/forum?id=BTKAeLqLMw.

Weiwen Liu, Xu Huang, Xingshan Zeng, Xinlong Hao, Shuai Yu, Dexun Li, Shuai Wang, Weinan Gan, Zhengying Liu, Yuanqing Yu, Zezhong Wang, Yuxian Wang, Wu Ning, Yutai Hou, Bin Wang, Chuhan Wu, Xinzhi Wang, Yong Liu, Yasheng Wang, Duyu Tang, Dandan Tu, Lifeng Shang, Xin Jiang, Ruiming Tang, Defu Lian, Qun Liu, and Enhong Chen. Toolace: Winning the api calling race with a comprehensive and high-quality dataset, 2024b. URL https://arxiv.org/abs/2409.00920.

Adyasha Maharana, Jaehong Yoon, Tianlong Chen, and Mohit Bansal. Adapt-∞: Scalable continual multimodal instruction tuning via dynamic data selection. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/a6610efd6c767f63343a4ab28505212e-Abstract-Conference.html.

Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. Gorilla: Large language model connected with massive apis, 2023. URL https://arxiv.org/abs/2305.15334.

Shishir G. Patil et al. The berkeley function calling leaderboard. In *Proceedings of Machine Learning Research*, 2025. URL https://proceedings.mlr.press/v267/patil25a.html.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. In *Advances in Neural Information Processing Systems*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/e6385d39ec9394f2f3a354d9d2b88eec-Abstract.html.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, Sihan Zhao, Lauren Hong, Ruobing Tian, Ruobing Xie, Jie Zhou, Mark Gerstein, Dahai Li, Zhiyuan Liu, and Maosong Sun. Toolllm: Facilitating large language models to master 16000+ real-world apis. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=dHng2O0Jjr.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=ryxGuJrFvS.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessi, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems*, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d842425e4bf79ba039352da0f658a906-Abstract-Conference.html.

Harsh Trivedi, Tushar Khot, Mareike Hartmann, Ruskin Manku, Vinty Dong, Edward Li, Shashank Gupta, Ashish Sabharwal, and Niranjan Balasubramanian. Appworld: A controllable world of apps and people for benchmarking interactive coding agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 2024. URL https://aclanthology.org/2024.acl-long.850/.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less: Selecting influential data for targeted instruction tuning. In *International Conference on Machine Learning*, 2024. URL https://arxiv.org/abs/2402.04333.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy Liang. Data selection for language models via importance resampling. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2302.03169.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains, 2024. URL https://arxiv.org/abs/2406.12045.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/3fe78a8acf5fda99de95303940a2420c-Abstract.html.

A Dataset and Split Provenance

The API-Bank train universe contains 6,839 valid records. CAMS uses only train-side partitions. The Stage-2 candidate pool, Stage-1 data, capability probe, and FCS confirmation surface are frozen before their respective evaluations and are disjoint under the documented exclusion lists. FCS contains 256 API-call records.

A.1 CAMS construction

The rare-frequency universe is the 6,839 converted valid train records. Each record contributes at most one count per called API. APIs are sorted by frequency, then stable API name; the first $\lfloor 1531/2 \rfloor = 765$ are rare, with the odd median assigned to common. This is a train-side exposure-scarcity definition, not a held-out frequency estimate.

The fixed probe contains 256 records, 64 each from development surfaces C2–C5. Capability correctness is averaged within each nonempty surface slice and then equally across surfaces. Probe outcomes enter the deficits, while confirmation outcomes are excluded from selection. The CAMS capability labels are computed from supervised gold calls. The multi-argument label is one when any valid gold call in the first supervised assistant message has at least two keyword arguments. The rare-API label is one when any gold call uses an API in the bottom 50% of the train-frequency distribution. In the eligible Stage-2 candidate pool, the demand-category counts are 122 both-demand records, 363 multi-only records, 106 rare-only records, and 1,425 neither-demand records.

For the three Stage-1 states, the measured deficits are:

State	D_{multi}	D_{rare}
0	0.6112	0.4438
1	0.7445	0.6125
2	0.6154	0.4688

Because all deficits are positive, both-demand records receive the highest CAMS score for all states. The both-demand pool has 134,121 serialized tokens, exceeding the 33k Stage-2 budget, so all three states select the same 32 records.

B Training and Evaluation Details

Core and pre-frozen baselines. The common Stage-1 core contains 32 records and 32,585 serialized tokens. Candidates are ranked by maximum cosine similarity to 512 train-only CORE_SUPPORT records, using normalized GTR-T5-base embeddings at revision `9801579ce813fb37541e6098dffd17959fffcd6e`. The support is selected mechanically from the 1,024 query-only records. TIS denotes the embedding-relevance baseline: maximum cosine to all 1,024 query-only records, descending score with record-ID ties. Stage 2 excludes the core and reuses this ranking.

The displayed LESS baseline is a local gradient-similarity implementation. It projects raw gradients of an initialized Qwen3 LoRA model into 128 dimensions with a deterministic signed hash, safely L2-normalizes candidate and query vectors, and ranks by maximum query cosine. Its frozen ranking is reused across states after excluding the core. It does not implement the original LESS warmup/checkpoint aggregation or Adam-preconditioned influence procedure; comparisons should be interpreted at this implementation scope. StaticAgreement reuses a minimum-across-environments percentile ranking; SCMA recomputes a state-conditioned projected-gradient ranking. The pre-frozen baselines appear in the main confirmation table; post-confirmation mechanism controls are reported separately.

Serialization and loss. API-Bank conversion creates a user context and a supervised assistant API-call answer. The first assistant-call answer plus EOS receives loss; prompt tokens are masked with -100. At maximum length 2,048, the prompt is left-truncated to retain the answer. If the answer alone reaches the limit, its first 2,048 tokens are used. Selection keeps records intact but does not imply untruncated context or multi-turn supervision. The table reports existing serialized-row metadata, not new instrumented training measurements.

Table 2: Actual Stage-2 exposure under the fixed serialized-token convention. Batch size one makes record count equal to continuation steps.

Subset	Records/steps	Serialized tokens	Supervised tokens
CAMS	32	33,027	935
Random-B1	43	33,154	747
Random-B2	38	32,741	803
Random-B3	44	33,227	826
TIS	32	33,379	748
LESS	90	32,993	1,287
StaticAgreement	91	33,046	1,440

Optimization and branching. Qwen3-8B uses BF16 LoRA, rank 8, alpha 16, dropout zero, no bias, and q/k/v/o projection targets. AdamW uses initial learning rate 2×10^{-4} with library defaults (betas 0.9/0.999, epsilon 10^{-8}, weight decay 0.01). Batch size and gradient accumulation are one, with one pass and no explicit gradient clipping. A linear zero-warmup schedule has a fixed 256-step horizon. Stage 2 restores both optimizer and scheduler from the corresponding Stage-1 state; it does not restart either. States use seeds 0, 1, 2. Three frozen Random subsets are crossed with the three states, producing nine models. Because record counts differ, total steps and final learning rates differ under this shared schedule.

The target total budget is 65,703 serialized tokens, giving a Stage-2 target of 33,118; CAMS actually uses 65,612 total tokens. Original subsets satisfy a 1.5% total-budget tolerance. CAMS selection ties use SHA256(record ID + `cams_tiebreak_20260905`); training order uses `scma_stage2_order_v1`. The fixed-order budget diagnostic is reported in Appendix G.3.

Evaluation. The train-side API-call evaluator sends the frozen system/user fields with non-thinking decoding, temperature 0.001 and maximum 512 output tokens. The parser extracts a bracketed function call and compares its name and keyword-argument dictionary to the gold call; malformed predictions are incorrect. This is parsed API-call exactness, not raw string equality, tool execution reward or interactive task success.

C Full FCS Results

Table 3: FCS API correctness per method. Random has nine runs; other continued selectors have three states.

Method	Scores								
Base	0.0625								
A-only	0.2617	0.2227	0.2422						
Random	0.3555	0.3047	0.3164	0.3438	0.3281	0.3242	0.3359	0.3125	0.3164
TIS	0.3320	0.3281	0.3438						
LESS	0.3125	0.3242	0.3516						
StaticAgreement	0.3398	0.3281	0.3203						
SCMA	0.3203	0.3008	0.2930						
CAMS	0.3438	0.3672	0.3477						

CAMS is compared with the mean of three Random subset proposals within each of three Stage-1 states. The FCS mean difference is 2.65 percentage points and its direction is positive in all three states. These states share the same 256 evaluation records and are not independent task replications. Model-level standard deviations in the main table describe variation among trained models, not uncertainty in the paired effect. We report no FCS-only confidence interval or significance claim.

Table 4: Underlying descriptive slice means, complementing the differences reported in the main text.

Surface	Method	Overall	Multi	Rare
FCS	Random	.3264	.4143	.5463
FCS	CAMS	.3529	.5143	.6806

These post-hoc FCS slice means do not establish uniform improvement across all record types.

D Construct Validity of CAMS Demands

These completed analyses explain CAMS's frozen construction. They reuse the 256-record train-side probe, existing Stage-1 and continuation predictions, TRAIN metadata, and frozen subset inventories. They do not retrain models, introduce selectors, change thresholds or membership, or use confirmation outcomes as construct-analysis targets. Cluster bootstrap intervals resample record IDs, retaining all three states' predictions; 20,000 draws use seed 20260921 with the recorded analysis-specific offsets. The provenance note below distinguishes historical diagnostic targets from the present construct audit.

D.1 Multi-binding structural demand

The frozen definition takes the maximum keyword-argument count over valid calls in the first assistant message. Positional arguments, unpacked arguments, and nonliteral keyword values fail the historical parser; nested values do not add bindings. Invalid calls are skipped. If none parses,

the operational count is zero, distinct from a measured zero-argument call. This affects 79 of 2,016 eligible candidates; their historical indicators are preserved. All 256 probe gold calls parse, with raw counts 0/1/2/3/4/5 supported by 5/190/32/21/6/2 records.

At ≥ 2, candidate support is 485/2,016 (24.06%) and probe support 61/256. The macro accuracy curve below is non-monotonic. Count versus record-level mean correctness has Spearman $\rho = 0.0440$ and Kendall $\tau = 0.0413$. The multi-minus-non-multi gap is $+10.05$ points (95% CI $[-1.84, 22.28]$), positive in all states; it does not support calling this slice harder. Thresholds ≥ 3 and ≥ 4 have candidate/probe counts 268/29 and 91/8, with gaps -5.01 and -25.94 points, respectively. Their CIs are $[-19.22, 10.69]$ and $[-31.05, -21.10]$; all eight high-count records are incorrect in every state. We retain the small group and all thresholds.

Table 5: Argument-bin exactness: equal mean across three states, record-cluster intervals. The eight-record group has a degenerate empirical bootstrap interval; this is not certainty about its population accuracy.

Arguments	Probe n	Accuracy (%)	95% CI
0-1	195	22.74	[17.44, 28.38]
2	32	43.75	[29.17, 58.33]
3	21	28.57	[11.11, 47.62]
4+	8	0.00	[0.00, 0.00]

API identity nearly determines argument count ($\eta^2 = 0.9971$; only 1/103 APIs varies within identity), and count correlates with log TRAIN frequency ($\rho = -0.5573$) and rare status ($\rho = 0.3455$). The state-adjusted count odds ratio is 0.952; adding frequency and query length yields 0.362, indicating composition sensitivity rather than causal identification. These observations support the structural meaning of simultaneous bindings but give weak marginal support for a difficulty interpretation. The ≥ 2 threshold is the minimal deterministic boundary for multi-binding tool-call structure, not an empirically optimal difficulty threshold.

D.2 Exposure scarcity and the frozen rare split

The bottom 765/1,531 TRAIN API identities account for 817/6,839 TRAIN records (11.95%), 232/2,048 pool records (11.33%), and 228/2,016 eligible candidates (11.31%). TRAIN frequencies count distinct record–API incidences; all observed TRAIN records have one distinct identity. This is library exposure, not measured optimizer exposure. The TRAIN Gini coefficient is 0.7028, consistent with concentrated use and a long tail, without asserting a power law. At the frequency-two boundary, 583 tied identities are split by the frozen name tie-break, 52 into rare and 531 into common.

Scarcity does not imply intrinsic difficulty. Frequency versus record-level mean correctness has Spearman $\rho = -0.5158$: lower-frequency APIs are more accurate on this probe. The threshold sensitivity table retains the four prespecified core splits; all twelve state-specific gaps are positive. At 50%, rare has 34 probe records and a $+27.55$-point macro gap, opposite to a harder-rare interpretation. Optional 10% and 80% references in the complete artifact have gaps $+25.07$ and $+37.78$ points, with probe support 2 and 62. No threshold is selected from these outcomes.

Table 6: Frozen-rank rarity sensitivity. Gap is rare minus common accuracy in percentage points, not a measure of scarcity.

Bottom %	APIs	Candidates	Probe n	Gap	95% CI
20	306	78	10	+18.94	[-8.81, +47.37]
33	505	137	16	+37.64	[+15.56, +58.06]
50	765	228	34	+27.55	[+12.11, +42.91]
67	1025	408	47	+37.85	[+24.90, +50.65]

The 50% split offers nonempty support while retaining a low-exposure identity region. Other ranks can also operationalize scarcity; 50% is a frozen, tuning-free median-rank discretization of exposure

scarcity, not a natural difficulty discontinuity or optimal cutoff. Surprisal $-\log p_{\mathrm{TRAIN}}(a)$ describes statistical scarcity; CAMS is neither entropy-maximizing nor diversity-maximizing.

Capability-selection scope. The complete screening artifact retains 18 historical metric/definition rows. Full exactness, parse validity, and argument-value correctness have no frozen intrinsic candidate demand; argument-key metadata alone does not supply a frozen demand formula. Tool presence is observable but constant over all 2,016 eligible records. Multi and current rare have nonempty support in all states. Long-query (852 candidates/128 probe records), current common (1,788/222), and later read/search, write, health, commerce, planning, and support slices remain possible alternatives outside the original instantiation. The evidence does not establish multi/rare as uniquely optimal. Strict predictive screening evidence from the historical C6/C7 correlations remains unavailable; the same-probe calibration in Appendix E answers a different question.

Historical diagnostic provenance. Historical Capability Response used a C6/C7-derived continuation target, so it is not strict train-side predictive screening evidence. Its legacy probe-defined rare universe differs from the current TRAIN-frequency demand; shared terminology does not make the definitions interchangeable.

E Residual-Need Calibration

The strict availability audit matches three Stage-1 states and seven existing continuation proposals to the same frozen probe, parser, slice definitions, and equal-surface aggregation. Current TRAIN-defined rare is remeasured consistently from existing predictions; old rare/common metrics and changing-denominator conditional-key metrics are excluded. There are 315 eligible state–capability–proposal pairs across 15 historical metrics; all 13 negative gains are retained. No confirmation-derived target enters this calibration.

For each pair, $D = 1 - C_{\mathrm{pre}}$, $\Delta = C_{\mathrm{post}} - C_{\mathrm{pre}}$, and normalized gain is $\Delta / \max(D, 10^{-12})$. Within multi and current rare, all seven proposals have positive deficit–gain Spearman coefficients: multi ranges from 0.5 to 1, rare from 0.5 to 1. Averaging the seven proposal gains equally gives $\rho = \tau = 1$ across the three states for each capability. These averages are summaries, not additional checkpoints or independent repetitions.

Table 7: Proposal-averaged calibration on the same probe; frozen equal-surface capability aggregation.

Capability	State	C_{pre}	D	C_{post}	Δ
Multi	0	.3888	.6112	.4500	.0612
Multi	1	.2555	.7445	.4752	.2197
Multi	2	.3846	.6154	.4620	.0773
TRAIN rare	0	.5563	.4438	.6643	.1080
TRAIN rare	1	.3875	.6125	.6688	.2813
TRAIN rare	2	.5313	.4688	.6732	.1420

For the six primary mean pairs, pooled raw-gain Spearman is 0.2571, within-capability-centered Spearman is 1.0000, and pooled normalized-gain Spearman is 0.0286. Across all 45 eligible capability–state means these values are 0.0013, 0.8933, and -0.4892. The primary capability-fixed-effect slope is 1.0648. Thus within-capability descriptive alignment coexists with weak pooled calibration and variable normalized-gain relationships.

The identity $\Delta = D + C_{\mathrm{post}} - 1$ couples deficit and gain through their common baseline; convergence of post scores and regression to the mean can contribute to positive association. Only three initial states are available, and no independent state validation was performed. The evidence is *moderate*: $1 - C$ is a bounded remaining-headroom proxy with partial descriptive alignment to future capability gain, not an independently validated future-value or per-unit efficiency predictor.

F SELECTOR BEHAVIOR AND SATURATION

The audit verifies the IDs, order, tokens, and provenance of all 24 historical method–state subsets. It recomputes $D_{\text{multi}}V_{\text{multi}} + D_{\text{rare}}V_{\text{rare}}$ using frozen deficits and demand semantics. The table reports all three normalizations, with enrichment relative to candidate record fractions. State averaging is an equal mean of three metric values, not pooling unequal SCMA subset sizes.

Table 8: Residual coverage and enrichment, averaged across states. Each normalization is reported; these deterministic full-subset summaries are not downstream outcomes.

Method	Mean	Total	Score/1k	Multi ×	Rare ×	Both ×
CAMS	1.1654	37.2915	1.1291	4.157	8.842	16.525
Random-B1	0.1848	7.9466	0.2397	0.870	0.823	0.769
Random-B2	0.3223	12.2483	0.3741	1.531	1.396	1.739
Random-B3	0.2071	9.1119	0.2742	0.945	1.005	1.127
TIS	0.4285	13.7111	0.4108	2.208	1.382	2.582
LESS	0.0202	1.8224	0.0552	0.092	0.098	0.184
StaticAgreement	0.0361	3.2851	0.0994	0.228	0.000	0.000
SCMA	0.1508	8.9027	0.2683	0.564	1.104	0.857

The candidate categories both/multi-only/rare-only/neither contain 122/363/106/1,425 records and 134,121/383,566/101,843/958,798 serialized tokens. The both pool is 4.05 times the 33,118-token budget. In states 0/1/2, $(D_{\text{multi}}, D_{\text{rare}})$ is $(.611218, .443750)$, $(.744471, .612500)$, and $(.615385, .468750)$, respectively, giving the same category ordering. All states select the same 32 both-demand records and 33,027 tokens. Their mean scores are 1.054968/1.356971/1.084135; totals are 33.758974/43.423077/34.692308; scores per 1,000 tokens are 1.022163/1.314775/1.050423.

In this regime CAMS reduces operationally to the frozen hash-ordered token prefix of the both-demand pool. The prefix undershoots by 91 tokens; adding the next 1,152-token record would overshoot by 1,061, so the nearest-prefix rule retains 32 records. State-specific deficits change scores but not membership. TIS and CAMS both contain 32 unique APIs with entropy 3.466, despite different residual coverage, demonstrating that API diversity and residual coverage are distinct diagnostics. This is implementation alignment, not a causal explanation of downstream gains or demonstrated state-adaptive membership.

Table 9: Original continuation subset structure. Structural statistics are descriptive, not a proxy for downstream success.

Subset	Records	Serialized tokens	APIs	Entropy
Random-B1	43	33,154	21	2.210
Random-B2	38	32,741	21	2.268
Random-B3	44	33,227	24	2.331
LESS	90	32,993	7	0.409
TIS	32	33,379	32	3.466
CAMS	32	33,027	32	3.466

G MECHANISM, EXPOSURE, AND BUDGET SENSITIVITY

G.1 FCS MECHANISM CONTROLS

We compare CAMS with four post-confirmation controls. MultiOnly prioritizes multi-argument records; RareOnly prioritizes rare-API records. CoverageMatchedRandom greedily seeks new APIs in hash-random order, targeting CAMS's 32 unique APIs before filling the budget. ShuffledDemand permutes paired capability labels across candidates before ranking. Each control uses one frozen subset and three training states. These are mechanism diagnostics on reused FCS, not new untouched confirmations.

Table 10: Mechanism controls on FCS. CAMS remains above ordinary Random, but coverage-matched and shuffled-demand controls are slightly higher in this saturated API-Bank regime.

Selector	FCS
Random	0.3264
CAMS	0.3529
MultiOnly	0.3086
RareOnly	0.3281
CoverageMatchedRandom	**0.3659**
ShuffledDemand	**0.3659**

Note: The controls are post-confirmation mechanism diagnostics. They do not change the CAMS–Random result on FCS under the frozen continuation recipe.

Both CoverageMatchedRandom and ShuffledDemand score above CAMS on FCS. Their actual composition is informative. CAMS selects 32 both-demand records covering 32 APIs. CoverageMatchedRandom selects 40 records covering 32 APIs, with only two both-demand records. ShuffledDemand selects 46 records covering 19 APIs, with one both-demand and 35 neither-demand records. Thus neither control should be described as reproducing CAMS's demand-rich subset. CAMS's improvement over ordinary Random remains, but these controls do not establish that accurate demand labels or measured deficits are necessary for the gain.

G.2 Supervision exposure accounting

For the separately frozen supervision-budget sensitivity study, Table 11 reports ranges across three subset proposals. Each proposal is trained from the same three historical Stage-1 states. These are training-input measurements, not effectiveness results. The target is 935 supervised tokens with whole-record nearest-prefix selection; matching supervision does not match sequence exposure, optimizer steps, or scheduler endpoints.

Table 11: Frozen Stage-2 exposure ranges across supplemental proposals. Counts exclude the shared Stage-1 prefix.

Selector	Supervised tokens	Serialized tokens	Steps
CAMS	935–953	29,272–33,279	27–31
Random	925–949	39,829–42,237	50–54
Coverage	934–949	39,073–42,858	47–51
ShuffledDemand	922–938	39,691–40,917	50–55

Complete per-phase selection and training timings were not recorded in the historical runs. We therefore do not infer GPU time from serialized tokens or file timestamps, and do not claim an end-to-end cost advantage. A complete comparison must include capability probing, embedding, selector-specific warmup and gradient collection, ranking, and SFT, with shared costs and their amortization stated explicitly.

FCS results under alternative budget contracts. These are post-confirmation diagnostics on reused FCS, with three subset proposals crossed with three historical Stage-1 states. The approximate supervised-token study targets 935 supervised tokens with whole-record selection. The following means are copied directly from the stored FCS method statistics; they are not inferred from a combined score.

Table 12: Existing FCS means under the approximate supervised-token budget.

Selector	FCS
CAMS	0.3346
Random	0.3446
Coverage	0.3594
ShuffledDemand	0.3481

CAMS remains below Random in this FCS study. The exact-label/update study admits exactly 935 consumed labels and 32 optimizer updates. OriginalCAMS denotes the historical demand-ranking rule under this alternative admission contract, not the original trained adapters. We report its recorded FCS model scores directly, without introducing a new aggregate or uncertainty estimate.

Table 13: Recorded FCS exactness for the exact-label/update diagnostic.

Proposal	State	OriginalCAMS	Random
0	0	0.3203	0.3516
0	1	0.3242	0.3359
0	2	0.3281	0.3594
1	0	0.3672	0.3555
1	1	0.3398	0.3594
1	2	0.3281	0.3555
2	0	0.3125	0.3320
2	1	0.3047	0.3164
2	2	0.3242	0.3359

Matching supervised tokens alone leaves record counts, updates, and scheduler endpoints unequal. Matching labels and updates still does not match unique supervision, context exposure, or forward-pass computation. Both studies vary subset proposals relative to the main experiment. They delimit the intervention being compared without identifying supervision volume as the cause of the FCS gain. The CAMS-Balanced variant is outside the main comparison.

G.3 Fixed-order budget sensitivity and LESS-port comparison

These are supplemental analyses on reused FCS, not independent confirmations. The budget experiment preserves the original CAMS and three Random orders, using nested prefixes and the same three Stage-1 states.

Table 14: CAMS–Random correctness differences under fixed selection orders, in percentage points.

Budget	FCS
8,192	-2.69
16,384	-0.04
24,576	+1.87

On FCS, the effect is negative at 8k and 16k, and positive at 24k. This is not uniform budget robustness or a supervision/compute-matched comparison. All budgets remain below both-demand supply.

Table 15: Three-state results for the LESS-port.

Surface	State 0	State 1	State 2	Mean
FCS	0.3516	0.3477	0.3438	0.3477

This LESS-port retains Adam-aware candidate gradients, SGD support gradients, and cross-checkpoint aggregation, using the recorded implementation. It is a supplemental algorithmic comparison, not a replication of the official LESS experiments or a replacement for the frozen LESS baseline.

H External Validity Boundary

Direct adapter portability. AppWorld direct transfer floors for the adapted models while Base remains nonzero. This is negative weight-portability evidence, not a clean test of selector portability in a matched training substrate.

Controlled STATE-derived comparison. This completed secondary matrix uses 19 frozen model conditions on 150 tasks with five runs each. All 14,250 expected outputs were scored, with no missing outputs, corrupt outputs, or configuration mismatches.

Table 16: STATE-derived Qwen3 V4 pass@1. Controlled corrective external, not official STATE, untouched evaluation, or leaderboard-comparable scoring.

Surface	Base	Random	CAMS	CAMS–Random
Overall	.145333	.124889	.102667	-.022222
Travel	.052000	.028444	.020000	-.008444
Customer support	.124000	.233778	.177333	-.056445
Shopping assistant	.260000	.112444	.110667	-.001777

Random averages nine models, CAMS three, and Base one. Overall means equally weight the three domains. CAMS underperforms Random overall and in every domain; its overall score also trails Base. We therefore treat this experiment as negative selector-portability evidence under this custom protocol.

Runner/evaluator corrections followed held-out access, so the result cannot be called untouched. All agent, simulator, and judge roles belong to the Qwen3 family; shared priors, stylistic preference, and judge calibration constrain external validity. The simulator audit also records reasoning-tag leakage in simulated user messages, further limiting fidelity to the intended user role. Our validity checks did not identify an implementation-invalidating error, but neither those checks nor the common backend isolate the contribution of these evaluator limitations. We therefore treat this result only as secondary negative portability evidence.

Artifact provenance. FCS was internally labeled C8 in historical artifacts. The accompanying repository retains frozen split and subset identities, parser contracts, model scores, serialized-token metadata, and per-record audit tables. The current evidence ledger maps each retained result to its source and records input hashes; historical records remain available separately. Demand audits use record-weighted bin accuracy, whereas CAMS deficits and calibration retain equal-surface aggregation. These measurement conventions are not interchangeable.