SamplerScope: Exact Decoder Attribution for Finite Language-Agent Behavior[1]

Ansh Dawda
Student, Universität des Saarlandes

With
Apart Research

Abstract

Studies often treat a sampled model action as evidence about the model, although decoding transforms the action distribution after logits are produced. SamplerScope isolates this inference-time contribution. We cached grammar-constrained action logits from two pinned Qwen2.5-Instruct checkpoints in two four-step Markov decision processes, permuted three one-token action labels exhaustively, and evaluated 11 decoder configurations on the same logits. Dynamic programming gave exact returns and outcome probabilities without rollout noise. Across 24 model-environment-label strata, greedy decoding improved return in 12 and reduced it in 12; changes ranged from -1.224 to +0.313 against raw softmax. Top-p at p=0.6 removed every optimal action over 18.5% to 73.3% of raw-policy decision occupancy, depending on the model and task. Reversing temperature and top-p changed return by as much as 0.275. Observed policies therefore belong to model-prompt-grammar-decoder systems, so behavioral studies should control and report decoding.

[1] Research conducted at the Digital Minds Research Sprint, August 2026

1. Introduction

Sampling is often treated as a deployment detail. That treatment is hard to defend when a language model acts in a sequential environment. A decoder does more than change the wording of one response: it changes the action distribution at each state, which changes later state occupancy and the final outcome. If an evaluation observes only completed trajectories, effects caused by the decoder can be misattributed to the model.

SamplerScope measures this contribution under controlled conditions. The model, prompt representation, valid actions, transition function, reward, and initial-state distribution stay fixed. We cache one set of valid-action logits per state and then intervene only on the decoder. Because the environments are finite and known, every induced policy is evaluated exactly rather than by Monte Carlo rollout.

Choice-based preference and welfare studies often treat a sampled action as evidence about the model. That attribution fails if the decoder changes the action. Here, weights and cached state logits stay fixed while the decoder changes action probabilities and downstream outcomes. Digital Minds studies should define the decoder as part of the target system or treat it as a measurement condition. They should test whether preference conclusions survive plausible decoder settings before attributing them to a checkpoint.

SamplerScope has two parts:

1. a small reusable pipeline from provenance-checked logit traces to decoder transformations and exact policy evaluation, and
2. a paired benchmark in which decoder effects are large, change sign, and interact with surface action labels.

The measured object is an operational action policy. It is not an intrinsic preference, a welfare claim, or evidence about sentience.

2. Related Work

Holtzman et al. [1], Nadeem et al. [2], and Nguyen et al. [3] study how nucleus, top-k, temperature, and min-p change open-ended generation. Dhamala et al. [4] show that decoder settings also change measured fairness. These studies score generated text. SamplerScope instead follows a fixed-logit decoder change through a policy with known transitions and rewards.

Ravfogel et al. [5] generate string counterfactuals under shared Gumbel noise; SamplerScope neither infers sampling noise nor claims a unique unit-level counterfactual. Aviary treats language agents as policies in decision processes [6]. We use that policy view in small, fully observed MDPs so values can be solved exactly. The contribution is the combination of grammar masking, decoder intervention, exhaustive label strata, and exact policy evaluation, not a new decoder or dynamic-programming algorithm.

Two agent studies connect decoder settings to downstream dynamics. DORA Explorer [9] uses sequence-level action sampling to improve exploration in bandits and text-adventure agents. De Nobili et al. [10] treat temperature as a control parameter in a single-token multi-agent Naming

Game. Both alter sampling and observe behavior. SamplerScope instead freezes state logits, applies each post-mask decoder offline, and solves the induced policy exactly in a known MDP.

3. Methods

3.1 Environment and Models

The benchmark contains two four-step environments. Service recovery has 18 reachable decision states and actions for backing up, repairing, or restarting a service. Queue control has 24 decision states and actions for adding capacity, serving normally, or using burst service. Each environment has success and failure terminal states. The state includes the round and all variables needed by the transition model, and the prompt states the exact transition and cost rules.

Each transition records success, failure, stakeholder cost, and one elapsed step. The scalar benchmark return is

$$return \;=\; success \;-\; 0.5 \;\times failure \;-\; 0.25 \;\times stakeholder\,cost$$

The outcome components are reported alongside the scalar return. This scalarization defines benchmark optimality; it is not offered as a welfare function.

We used pinned BF16 checkpoints of Qwen2.5-0.5B-Instruct and Qwen2.5-1.5B-Instruct [7]. For each of the 42 decision states, semantic action order stayed fixed while the one-token labels A, B, and C were exhaustively permuted. The resulting 504 local forward passes cover 42 states, six mappings, and two models. Every decoder comparison within a stratum reuses the same cached logits. Checkpoint revisions, digests, prompts, token IDs, and runtime versions are stored in the traces.

3.2 Decoder Intervention

The model produces full-vocabulary next-token logits $z(s)$. A grammar mask keeps the three valid action labels. A decoder D then induces

$$\pi_D(a \mid s) \;=\; normalize(D(mask(z(s))))$$

We evaluated raw softmax, greedy, two temperatures, top-k, two top-p settings, two min-p settings, and both orders of temperature 0.5 with top-p 0.8. The full grid is in Figure 1. Float64 reference semantics retain top-k cutoff ties and use token ID to resolve greedy and top-p score ties. The reversed composition is a counterfactual stack, not a Transformers default [8].

3.3 Exact Policy Evaluation

For horizon H, transition model P, and transition reward r, we compute

$$V_t^{\,D}(s) \;=\; \sum_a \pi_D(a \mid s)\;\left[\, r(s,a) \;+\; \sum_{s'} P(s' \mid s,a)\, V^{D}_{\,t+1}(s') \,\right]$$

The same recursion gives terminal hit probabilities, stakeholder cost, and state occupancy. Diagnostics include distributional distortion, mass removed by truncation, and whether a decoder

removes every co-optimal action. We use $KL(decoded \,||\, raw)$, since support truncation can make the reverse direction infinite.

The six label mappings are exhaustive fixed strata, not independent samples. Results are therefore descriptive and paired within strata. We report all signs and ranges rather than p-values or confidence intervals. Synthetic high- and low-margin logit controls test the evaluator against known optimal policies. Greedy recovered the exact optimum under both margins. Raw-softmax gaps were below 0.0021 at high margin and 0.357 to 0.378 at low margin.

4. Results

4.1 Decoder Effects Change Sign

Across 24 model-environment-label strata, decoder gaps relative to raw softmax ranged from -1.224 to +0.313 return units. Greedy helped 12 and harmed 12. At the two extremes, it moved 0.5B service recovery from 0.494 to -0.730 and 0.5B queue control from -0.235 to 0.078, both under mapping B-C-A.

Table 1 reports means across the six mappings. Greedy raised 0.5B queue-control return by 0.153 but lowered 0.5B service-recovery return by 0.424. It also lowered both 1.5B means. Top-p at 0.6 removed every optimal action over 18.5% to 73.3% of raw-policy decision occupancy, depending on the model and environment.

Table 1. *Exact return means over six label mappings. Censor is the raw-policy decision occupancy where top-p 0.6 assigned zero probability to every benchmark-optimal action.*

Model	Environment	Optimal	Raw	Greedy	Top-p .6 censor
0.5B	Queue control	0.121	-0.128	0.025	18.5%
0.5B	Service recovery	0.980	0.491	0.067	49.5%
1.5B	Queue control	0.121	-0.288	-0.298	34.2%
1.5B	Service recovery	0.980	0.566	0.441	73.3%

No decoder ranked first throughout. Temperature 1.5 improved mean service recovery for both sizes, yet in 0.5B queue control temperature 0.5 gained 0.055 while temperature 1.5 lost 0.030. A single decoder ranking would hide this interaction.

4.2 Surface Labels Moderate Decoder Effects

Changing the A/B/C assignment changed the logits even though semantic action order stayed fixed. Raw-policy return ranges across the six mappings were 0.227 to 0.445 in the four model-environment cells. Greedy ranges were still larger, from 0.349 to 1.670. Within a mapping, the dominant surface label accounted for an average 85.4% to 95.8% of raw greedy winners across the four cells. This is weak argmax state responsiveness, not evidence that the model ignored state, since non-argmax probabilities still changed.

4.3 Operator Order and Grammar Diagnostics

Temperature and top-p did not commute. Reversing temperature 0.5 and top-p 0.8 changed same-stratum return by as much as 0.275. Across the 24 paired strata, the direction was mixed: eight deltas were positive, eleven negative, and five effectively zero.

A valid action label ranked first in the full vocabulary in all 504 rows, so the grammar did not rescue arbitrary tail tokens. Mean pre-mask valid-action mass ranged from 0.929 to 0.9995. Exact ties occurred in 54 rows, making the documented tie rule necessary.

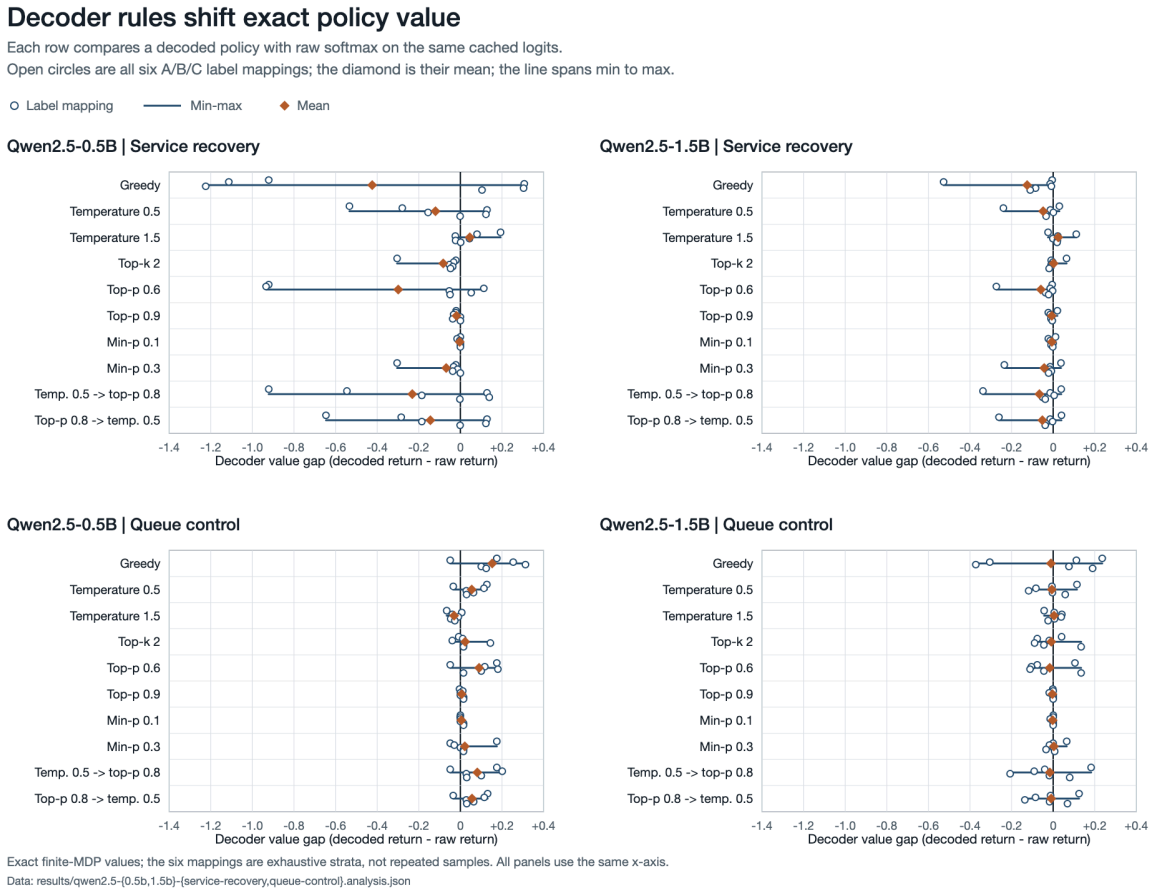

Figure 1. *Decoder value gaps relative to raw softmax on the same cached logits. Open circles show all six exhaustive label mappings, diamonds show their means, and lines span the fixed mapping range. These are strata, not repeated samples.*

5. Discussion and Limitations

Greedy and truncation were neither uniformly helpful nor harmful. Their rank changed across environments and label mappings. The stable result is attribution: with model logits fixed, the decoder changed occupancy, outcomes, and return. The operational policy belongs to the deployed model-prompt-grammar-decoder system.

Limitations

That conclusion is narrower than a claim about preferences. The benchmark uses two small checkpoints from one family, two handwritten MDPs, one prompt family, and a one-token action grammar. Surface-label effects and weak argmax state responsiveness rule out a strong planning interpretation. Different reward weights could also change the benchmark-optimal action, so outcome components remain part of every result. The float64 decoder reference is not claimed to be bit exact at device-specific threshold boundaries.

The paired logits, exact transitions, exhaustive labels, and known-logit controls make the reported sign reversals independent of rollout noise.

Future Work

The next tests should add unrelated model families, multi-token or tool-call actions, and alternative reward weights. Fixed-logit traces and paired policy evaluation should remain, or model and decoder changes become confounded.

6. Conclusion

SamplerScope treats decoding as part of a deployed language-agent policy rather than a neutral sampling afterthought. In two finite case studies, decoder-only interventions changed exact return substantially, sometimes in opposite directions under different environments or action labels. The included traces and evaluator make those differences auditable. They also make the scope plain: the project measures operational policies under a specified grammar and reward, not intrinsic preferences or welfare.

Code and Data

- Code repository: SamplerScope on GitHub
- Data and artifacts: the repository's results/ directory contains all four logit traces, the corresponding analyses, and synthetic controls. Each analysis records the SHA-256 of its source trace.

References

1. A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi. 2020. The Curious Case of Neural Text Degeneration. ICLR 2020. OpenReview

2. M. Nadeem, T. He, K. Cho, and J. Glass. 2020. A Systematic Characterization of Sampling Algorithms for Open-ended Language Generation. AACL-IJCNLP 2020, 334-346. DOI

3. N. M. Nguyen, A. Baker, C. Neo, A. G. Roush, A. Kirsch, and R. Shwartz-Ziv. 2025. Turning Up the Heat: Min-p Sampling for Creative and Coherent LLM Outputs. ICLR 2025. OpenReview

4. J. Dhamala, V. Kumar, R. Gupta, K.-W. Chang, and A. Galstyan. 2023. An Analysis of the Effects of Decoding Algorithms on Fairness in Open-Ended Language Generation. IEEE SLT, 655-662. DOI

5. S. Ravfogel, A. Svete, V. Snaebjarnarson, and R. Cotterell. 2025. Gumbel Counterfactual Generation From Language Models. ICLR 2025. OpenReview

6. S. Narayanan et al. 2024. Aviary: Training Language Agents on Challenging Scientific Tasks. arXiv:2412.21154. DOI

7. Qwen Team. 2024. Qwen2.5 Technical Report. arXiv:2412.15115. DOI

8. Hugging Face. 2026. Transformers generation utilities. Hugging Face documentation

9. P. Gurjar, M. F. Ishmam, and K. Marino. 2026. DORA Explorer: Improving the Exploration Ability of LLMs Without Training. arXiv:2604.17244. DOI

10. C. De Nobili, V. Iyer, A. Codello, and R. Burioni. 2026. Microscopic Dynamics of Consensus Formation in Multi-Agent LLM Naming Games. arXiv:2608.02178. DOI

Appendix

Limitations and Dual-use / Ethical Considerations

This benchmark does not measure consciousness, sentience, subjective welfare, or a stable latent utility function. It uses operational action probabilities in synthetic environments. The results should not be used as evidence that a model has interests or that a decoder reveals a model's "true" preferences.

The main technical limitations are the two Qwen2.5 checkpoints, two synthetic MDPs, one state-prompt format, one-token A/B/C actions, and a four-step horizon. All decoder interventions occur after a valid-action mask, so the results do not directly describe ordinary free-form generation. The six label mappings are exhaustive controls, not a statistical sample. The reported variation therefore has no population-level confidence interval.

Decoder audits have a dual use. They can find deployments where truncation silently removes safer or benchmark-optimal actions. The same machinery can also tune a decoder toward an operator's chosen reward without changing model weights. Because any scalar reward encodes a normative choice, an audit should publish its outcome components, masks, and decoder configuration rather than only the optimized score. SamplerScope stores these details and keeps the raw policy as a paired baseline.

The study used open-weight checkpoints locally and no personal data or paid model API. The prompts and traces contain only synthetic environment states.

LLM Usage Statement

OpenAI Codex assisted with implementation, test generation, plotting, literature discovery, and report drafting. I designed the study, made the methodological decisions, verified the citations, and independently recomputed every reported result from the saved traces.