Position: Beyond the Single Solution — The Rashomon Effect in Reinforcement Learning

Sourav Panda[1], **Aviral Srivastava**[2*], **Jonathan Dodge**[1]

[1]College of Information Sciences and Technology, Pennsylvania State University
[2]Amazon
sbp5911@psu.edu, aviralsr@amazon.com, jxd6067@psu.edu,

Abstract

This paper extends the Rashomon Effect to Reinforcement Learning (RL), leveraging it as a framework for analyzing behaviorally diverse yet performance-equivalent policies. We begin by formalizing analogies: between datasets and environments, between losses and rewards. These analogies let us define the *Rashomon set of RL agents* as the set of policies that achieve comparable returns while differing in behavior. This framing highlights multiplicity as an inherent property of learning rather than stochastic noise, with implications for alignment, interpretability, and retraining. We further extend the concept to multi-criteria settings, showing how multiple overlapping equivalence criteria reveal structured diversity within policy spaces. Viewing RL through the Rashomon lens encourages systematic study of behavioral multiplicity as a foundation for more robust, interpretable, and human-aligned agents.

Introduction

The Rashomon Effect in Machine Learning The *Rashomon Effect*, introduced by Breiman in *Statistical Modeling: The Two Cultures* (Breiman 2001), highlights that many distinct models can fit the same data similarly well. Inspired by the film *Rashōmon* (Kurosawa 1950), it captures how multiple plausible yet conflicting explanations can coexist. Building on this idea, (Fisher, Rudin, and Dominici 2019) formalized the *Rashomon set*—the collection of models whose predictive performance lies close to the best-performing model. The Rashomon Effect provides an analytical framework on model multiplicity which has proven effective in supervised settings (Dong, Rudin, and Seltzer 2020; Marx, Calmon, and Ustun 2020; Semenova, Rudin, and Parr 2022; Rudin, Semenova, and Parr 2024). While these studies establish the Rashomon framework as a powerful lens for understanding model diversity, they remain grounded in settings with a *fixed dataset* and consistent evaluation metric. How such multiplicity extends beyond fixed-data paradigms remains largely unexamined.

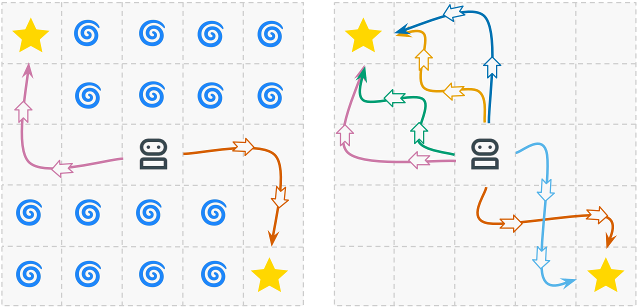

Figure 1: Gridworld examples illustrating behavioral diversity in RL. Cells marked with spirals represent inaccessible regions. (**Left**) A simple environment with two optimal routes to identical rewards. (**Right**) An environment with higher state connectivity and dynamic complexity, where multiple distinct trajectories achieve the same outcome. **Takeaway:** Complexity ↑ ⇒ Behavioral diversity ↑

From Fixed Data to Interactive Learning Reinforcement learning (RL) challenges the foundational fixed-data assumption. Unlike supervised settings, where all models learn from the same dataset, each RL agent either: (1) *generates its own individualized experience* through on-policy interaction with an environment that may or may not be stochastic; or, (2) in off-policy formulations, learns from trajectories generated by different behaviors. Different exploration trajectories expose agents to distinct subsets of the state–action space, effectively giving rise to different data distributions. As shown in Figure 1, even when agents are trained under identical reward structures, they may arrive at comparable returns through distinct state visitation patterns or strategies. In simpler environments Figure 1(left), the set of near-optimal behaviors is limited—only a few paths lead to the same reward—whereas in more complex, less constrained environments Figure 1(right), the space of

Position: The Rashomon Effect naturally extends to RL and is useful to study *behavioral multiplicity* within the environment, just as it studies *model multiplicity* on fixed data.

[*]This work was conducted independently. Amazon was not involved in this research.

1st Workshop on Navigating Model Uncertainty and the Rashomon Effect: From Theory and Tools to Applications and Impact

equally good trajectories expands substantially. This scaling from a small to a large set of behaviorally diverse solutions illustrates that, as environments grow in complexity, equally successful agents can exhibit markedly different behaviors—diversity that warrants systematic study rather than being treated as noise.

Formulating the Rashomon Framework in Reinforcement Learning

We formalize the *Rashomon framework* for RL by drawing analogies between the fixed elements of classical Rashomon framework and those in interactive learning. Specifically, we relate datasets to environments and losses to rewards, introducing the notion of a *Rashomon set* that captures behavioral multiplicity among comparable-performing policies.

Fixed Foundations: Environment and Reward

In supervised learning, the Rashomon Effect arises when many models achieve comparable predictive performance on the same fixed dataset. The dataset provides a common source of evidence, while the loss function defines a uniform criterion for evaluating model quality. Extending this framing to RL requires identifying analogous constructs.

Environment $\approx$ Dataset. In classical supervised learning, the entire dataset is available to the model, with the goal of approximating the data-generating function that is the origin of the samples comprising the dataset. Each model implicitly determines which features or relationships to emphasize while fitting the same examples.

Consider the case of a *Markov Decision Process (MDP)*, which represents one of the most common formulations where RL is applied, though RL can extend beyond the MDP setting. An MDP environment $\mathcal{E}$ can be formally defined as shown in Equation 1:

$$\mathcal{E} = \{\mathcal{S}, \mathcal{A}, P, R, \rho_0, \gamma\}, \tag{1}$$

where: $\mathcal{S}$ is the set of possible states, $\mathcal{A}$ is the set of possible actions, $P(s'|s,a)$ is the transition probability, $R(s,a)$ is the reward function, ρ_0 is the initial state distribution, and $\gamma \in [0,1)$ is the discount factor. Equation 1 thus specifies the underlying data-generating process: it defines the complete space of possible trajectories that any policy could, in principle, experience through interaction.

While every agent interacts with the same underlying environment, each policy π samples from it differently, generating its own distribution over transition tuples (s, a, r, s') based on its exploration behavior and learning dynamics. This variation in *data acquisition* parallels how different predictive models emphasize distinct features or relationships within the same dataset.

Thus, in the RL Rashomon framework, the environment serves as a shared but interactively accessible data source—a fixed foundation from which multiplicity arises through diverse trajectories of experience. Even in off-policy settings, where agents learn from replay buffers or trajectories from other polices, this diversity persists: the underlying environment remains constant, but the observable data distribution shifts according to which policies generated the experience.

Reward $\approx$ Loss. In supervised learning, the loss function provides a scalar measure of model performance with respect to a fixed dataset. All models optimize the same loss function (e.g., mean squared error or cross-entropy) over identical samples and labels. This shared loss evaluates all models against the same evidence and criterion, allowing differences in performance or structure to reflect their inductive biases rather than differences in feedback.

In RL, the reward function $R(s,a)$ plays an analogous role by defining success for an agent. The resulting expected return serves as the objective for optimization across all policies, shown in Equation 2.

$$J(\pi) = \mathbb{E}_{\tau \sim p_\pi}\left[\sum_{t=0}^{T} \gamma^t R(s_t, a_t)\right], \tag{2}$$

However, unlike the loss in supervised learning—which evaluates models on a fixed dataset—the reward in RL comes from interaction with the environment and is therefore inherently *policy-dependent*. Each policy π generates its own trajectory distribution $\rho_\pi(s,a)$, shaping the feedback it receives. Thus, while the reward function provides a consistent measure of success analogous to the loss, Equation 2 captures outcomes derived from each policy's unique experiences rather than a shared static dataset.

Holding the environment and reward function constant establishes the fixed foundation necessary for defining multiplicity in RL. The environment provides the shared evidence base, while the reward—often the most succinct and direct representation of the task itself (Ng and Russell 2000)—defines the objective of optimization. Together, they play roles analogous to the data-generating function and loss in classical Rashomon framework, forming a stable ground from which diverse policies can emerge as equally competent yet behaviorally distinct solutions.

Defining a Rashomon Set for RL Agents

In parallel with the classical setting, we begin with the most direct instantiation of a Rashomon set in RL: one defined by the reward objective. Formally, the *Rashomon Set* in RL is the set of policies that achieve similar expected returns within the same environment, as shown in Equation 3:

$$\mathcal{R}_\epsilon^{RL} = \{\pi \in \Pi \mid J(\pi^*) - J(\pi) \leq \epsilon\} \tag{3}$$

where π^* denotes a reference optimal policy and ϵ specifies the near-optimality tolerance. Unlike the symmetric definition used in the classical Rashomon set—where model performance may vary above or below the optimum by $\pm\epsilon$—the RL case is inherently one-sided. Since $J(\pi^*)$ represents the maximum achievable expected return under the fixed reward function, deviations can only occur in the negative direction. Hence, the RL Rashomon set contains all policies whose expected return lies within ϵ of the optimal value, i.e., $J(\pi) \in [J(\pi^*) - \epsilon, J(\pi^*)]$. Policies within $\mathcal{R}_\epsilon^{RL}$ share comparable performance despite the high likelihood of differences in experience and behavior.

Unlike the classical-Rashomon set, which measures multiplicity across models trained on a fixed dataset, we define the RL-Rashomon set over trajectory distributions $\tau \sim$

$p_\pi(s, a)$ arising from interaction with the environment. Here, behavioral multiplicity replaces feature multiplicity: different policies can achieve the same expected return while emphasizing distinct regions of the state–action space or adopting different exploration patterns. This formulation establishes the simplest form of multiplicity in RL—based purely on the scalar reward—but also motivates a broader view of what criteria can define equivalence.

Beyond Return Equivalence: Defining Rashomon Sets for Behavioral Diversity

While the definition above mirrors the classical Rashomon formulation, equating comparable-performance with similar expected return introduces a fundamental limitation in the RL context. In supervised learning, evaluation metrics such as accuracy or cross-entropy are directly tied to task performance; a model with lower loss or higher accuracy is unambiguously "better" within the defined objective. These metrics are complete with respect to the dataset—they fully capture the notion of success. As a result, numerical similarity alone defines the Rashomon set in classical ML, without requiring post hoc interpretation of the evaluation metric.

In contrast, RL decouples numerical reward from true task performance. Policies that achieve similar or even maximal return can differ drastically in their qualitative behavior. Some may exploit loopholes in the reward function—a phenomenon often referred to as *reward hacking* (Amodei et al. 2016)—achieving high cumulative reward while performing poorly according to human judgment or task intent. Thus, reward similarity alone does not guarantee behavioral or semantic similarity across policies.

Unlike classical supervised learning—where fixed data and loss metrics comprise evaluation—RL admits inherently richer *single-criterion* formulations that arise from the dynamics of interaction. These criteria often have no direct analogue in classical ML because they depend on how an agent *acts over time*. Formally, such sets are expressible as shown in Equation 4:

$$\mathcal{R}_\epsilon^m = \{\pi \in \Pi \mid m(\pi_m^*) - m(\pi) \leq \epsilon\}, \tag{4}$$

where $m(\pi)$ measures one aspect of policy performance beyond return and π_m^* denotes the best performing policy with respect to metric $m(\cdot)$. The metric $m(\cdot)$ can capture behavioral aspects of a policy that are not reflected in its return, offering alternative ways to evaluate how an agent achieves its outcomes. These aspects may relate to temporal dynamics, stability, or exploration behavior, among others. The following examples illustrate several such criteria.

(a) Temporal Efficiency. Agents can achieve the same reward while differing in how quickly or directly they reach their goal. One agent might take a long, cautious route; another might find a shorter or more aggressive path. This notion of efficiency—the expected time or number of steps to success—captures both the temporal cost and the behavioral path of an agent. It has no direct analogue in static supervised settings, where task completion occurs in a single evaluation step.

(b) Consistency under Uncertainty. Policy optimization in RL occurs *in expectation*—agents maximize the expected return rather than any single observed outcome. Consequently, two policies may attain similar expected performance yet differ in the stability of that performance across episodes. One may exhibit low variance, following a risk-averse strategy that performs reliably, while another may display high variance, pursuing a high-risk, high-reward strategy that alternates between failure and overperformance. Equivalence defined through measures of reliability captures an RL-specific notion of quality—one that makes a critical difference when policies are deployed in high-stakes or safety-sensitive settings.

(c) Exploration Preference. Agents can also differ in how they balance exploration and exploitation. An exploratory agent may continue sampling unfamiliar states even after finding a good policy, while a more conservative agent exploits known high-reward actions. Both can achieve comparable long-term return, yet their behaviors, data distributions, and learned representations are markedly different. Exploration tendency therefore defines another single-criterion Rashomon set that cannot arise in classical supervised learning, where the dataset is complete and originates externally.

These examples illustrate that even single-criterion Rashomon sets in RL encode dimensions of performance rooted in interaction—time, uncertainty, and control stability—none of which have direct analogues in supervised learning. They arise naturally from the agent–environment loop rather than from algorithmic augmentation, underscoring that multiplicity in RL is an *inherent* property of learning through experience rather than a product of explicit diversity design.

In this view, the single-criterion Rashomon set in RL provides a multidimensional surface of behavioral diversity beneath an apparently scalar objective. Expected return similarity marks only one axis; many others emerge naturally from the agent's interaction with the environment. This interactive grounding differentiates the RL Rashomon landscape from its supervised counterpart and motivates extending the framework to multiple criteria, as we discuss next.

From Single to Multi-Criteria Rashomon Sets

The richness of single-criterion RL Rashomon sets naturally raises a question: why stop at one criterion? Real learning problems rarely optimize a single metric—they balance multiple objectives. If a Rashomon set captures multiplicity under one measure of comparable-performace, it is generalizable to describe multiplicity under multiple, potentially interacting criteria. This generalization is conceptually analogous to Pareto fronts in multi-objective optimization, where evaluation of solutions occurs jointly rather than hierarchically (Deb and Kalyanmoy 2001; Van Moffaert and Nowé 2014).

Extending this view beyond RL, the notion of multi-criteria Rashomon sets applies to *learned functions* in general, not only to policies. In classical ML, this enables simultaneous consideration of competing objectives (e.g., ac-

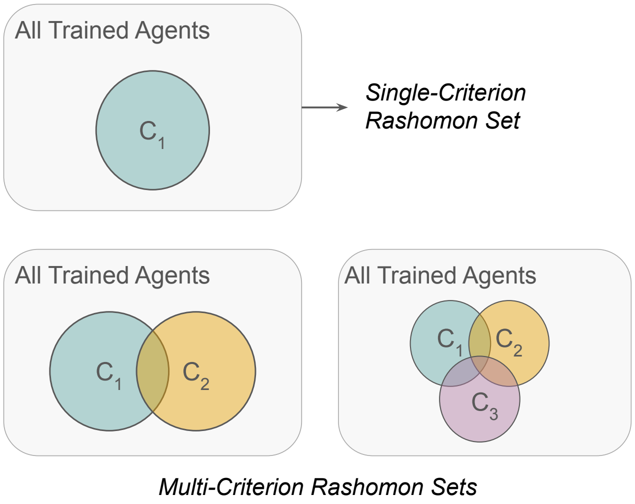

Figure 2: Illustration of Rashomon sets extending from a single performance criterion (C_1) to multiple, overlapping criteria (C_1, C_2, C_3).

curacy, precision, recall, or F1-score) offering a richer picture of model multiplicity than single-metric formulations. In RL, this perspective becomes even more critical: the relevant criteria often extend beyond numerical performance to encompass qualitative aspects of behavior—(e.g., stability, risk sensitivity, or exploration) By treating these interacting criteria as overlapping Rashomon sets, multiplicity becomes a structured property of learning dynamics rather than a byproduct of noise.

Formally, one can write a multi-criteria Rashomon set as shown in Equation 5:

$$\mathcal{R}_{\boldsymbol{\epsilon}} = \{\, f \in \mathcal{F} \mid m_i(f^*_{m_i}) - m_i(f) \leq \epsilon_i, \\ \forall\, i \in \{1, \ldots, K\}\,\}. \tag{5}$$

where each m_i is a performance metric, ϵ_i its tolerance, and $f^*_{m_i}$ is the best performing learned function with respect to criterion $m_i(\cdot)$. This formulation extends the Rashomon framework from scalar to vector-valued evaluation. Figure 2 illustrates this generalization.

The relationships among these sets—such as their intersections and unions—capture how different criteria interact or compete, a topic elaborated in the following section.

Implications and Analytical Perspectives

Having established the conceptual formulation of a Rashomon set in RL, we now consider its broader implications. The following discussions examine how acknowledging multiplicity reshapes our understanding of policy diversity, evaluation, and interpretability in RL, and what new perspectives emerge when treating variability as an informative property of learning rather than a source of noise.

Preserving Behavioral Diversity

Having identified a Rashomon set of RL agents—comprising policies that achieve comparable expected returns within a shared environment—the next question is how to analyze such multiplicity. A common practice is to aggregate predictions from top-performing models, for example by averaging outputs or ensembling parameters, to improve stability and generalization. However, prior work shows that such aggregation can diminish model diversity and lead to homogenized behavior or "learner collusion" (Wood et al. 2023; Jeffares et al. 2023). In case of RL, ensemble-based approaches seem to reduce exploration diversity and mask distinct strategies (Lin et al. 2024).

Each policy within the Rashomon set potentially represents a distinct way of solving the same task—a unique allocation of attention, exploration pattern, or prioritization of subgoals. Collapsing these policies into a single representative model may improve stability, but it erases the behavioral variability offered by the Rashomon set. Preserving individuality therefore becomes central: we should view each near-optimal policy as an independent sample from the broader solution space. Comparing these policies reveals how different inductive biases, exploration dynamics, or local optima yield diverse yet successful strategies—diversity that reflects an intrinsic property of the learning process rather than noise to be optimized away.

Alignment as Selection within the Rashomon Set

The existence of multiple comparable performing policies raises a natural question of alignment: if several agents satisfy the same training objective, which among them best reflects the preferences or intentions of their designers or users? Within RL, practitioners typically treat the reward function as the ground truth specification of a task (Ng and Russell 2000), despite knowing that it only approximates desired behavior. Different policies can fulfill the same reward criterion while exhibiting distinct behavioral characteristics that may align differently with human expectations. A similar situation arises in sports: two basketball players may score the same number of points, one relying on consistent two-pointers and another on riskier three-point shots. Both achieve comparable success by the game's metric, yet their styles differ in stability and risk—mirroring how distinct RL policies can meet the same reward but reflect different strategic preferences.

This issue extends beyond reward-based definitions of the Rashomon set. Defining similarity along other criteria still raises alignment concerns. Multiple policies may achieve comparable scores on these auxiliary dimensions while differing in trade-offs that are ethically or operationally significant. Hence, we can understand preference alignment as the process of navigating among Rashomon sets from different criteria, each reflecting a particular view of what constitutes "good" performance. Figure 3 concretizes the argument by mapping the abstract set operations from Figure 2 onto a grid-world example with three criteria (star, diamond, circle), showing how different intersections correspond to distinct policy behaviors.

From this perspective, alignment becomes a problem of *selection* rather than *retraining*. Instead of modifying the objective, one can explore the landscape of existing com-

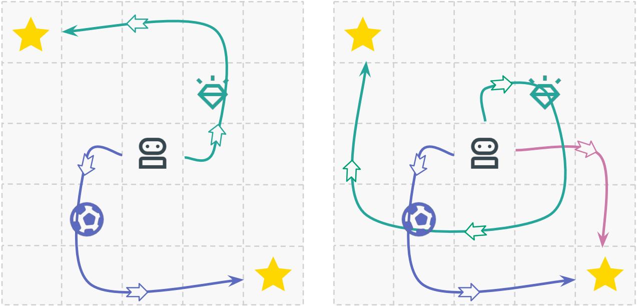

Figure 3: Preference extraction from a Rashomon set with three criteria. **Left:** Two policy trajectories, each satisfying exactly two of the three criteria. **Right:** Three trajectories satisfying different numbers of criteria—one meets a single criterion, one meets a pair, and one meets all three.
Takeaway: Each policy path lies in one of the intersecting regions of the multi-criteria Rashomon space, corresponding to specific combinations of alignment among the three objectives.

petent policies and identify those that align best with human values, safety requirements, or contextual norms. This interpretation also complements recent approaches such as *reinforcement learning from human feedback (RLHF)* (Christiano et al. 2017), where human input implicitly selects among near-optimal behaviors rather than explicitly defining the underlying reward. Viewing alignment through the Rashomon lens highlights how to leverage diversity in the solution space to better match learning systems to human intent.

Rashomon-Guided Retraining and Design Iteration

A common response to unsatisfactory model behavior—whether in RL or other learning paradigms—is to modify the objective and retrain. This cycle often repeats until the observed performance aligns with designer expectations. In some cases, a domain expert may adjust specific reward or loss components based on human knowledge about the task without knowing whether these modifications will sufficiently redirect the model's behavior. In other cases—where such expertise is limited—the process becomes largely based on intuition, relying on guesswork rather than evidence. Both situations lead to the same bottleneck: a trial-and-error loop of retraining without a clear understanding of what behavioral changes each modification actually induces.

The Rashomon framework offers an alternative approach. Rather than retraining reactively after each disappointing outcome, one can begin by deliberately training a set of agents that explore diverse behavioral solutions under the same objective. This collection—the Rashomon set—represents the space of comparable performing policies but through different behavioral strategies. Examining these policies provides empirical insight into how the same objective admits multiple interpretations. Instead of asking "what new reward might fix this behavior," designers can ask "which existing behavior already aligns best with the intended goal?"

Viewed this way, the effort of training n agents is not wasted exploration. In the conventional loop, n retraining attempts occur blindly, each a guess about what adjustment might work. In a Rashomon-guided process, those same n training runs are structured intentionally, producing a diverse set of interpretable policies that reveal trade-offs across known criteria. This transforms retraining from an uncertain search into an evidence-guided iteration process. Figure 4 illustrates this contrast, comparing the conventional trial-and-error retraining loop with the Rashomon-guided process that incorporates an explicit analysis stage.

Importantly, the goal is not to eliminate retraining but to replace guess-based iteration with a more informed and purposeful process. The Rashomon set functions as an empirical reference point—a repository of viable solutions that exposes some of the behavioral degrees of freedom under the current reward specification. This framing encompasses practices in modern AI systems such as: league training frameworks (Vinyals et al. 2019), which co-evaluates multiple policies to inform targeted improvements and avoid cycle chasing behavior; hypergrid search, which explores diverse configurations to reveal performance plateaus; and cross-validation, which leverages variation across folds to assess generalizable behavior. Acting with such structured insight, rather than in ignorance, enables designers to reason about what the agent *could* do—not just what it *did*—providing a more principled basis for iterative refinement.

Conceptual Background and Related Work

Although prior research in RL, interpretability, and ensemble methods has examined phenomena related to multiplicity, none has explicitly formalized them as a Rashomon setting in RL. Rather than providing an exhaustive literature review, this section traces the conceptual ancestry of the idea. We highlight how multiple disconnected threads have implicitly studied the same underlying property—the existence of behaviorally distinct yet equally competent agents.

Behavioral Diversity and Skill Discovery in RL

Several strands of research actively investigate how diverse behaviors emerge in RL agents. Methods such as DIAYN (Eysenbach et al. 2019) and MLSH (Frans et al. 2017) deliberately optimize for skill diversity by embedding information-theoretic or hierarchical objectives into the reward structure. Population-based and ensemble approaches—including Bootstrapped DQN (Osband et al. 2016) and Evolution Strategies (Salimans et al. 2017)—train multiple agents in parallel, producing distinct yet comparably performing policies that differ in exploration dynamics and representational bias. Although these approaches successfully elicit behavioral diversity, they do so by *design*—through algorithmic modifications or explicit diversity incentives. The Rashomon perspective, in contrast, focuses on *inherent multiplicity*: the spontaneous coexistence of many equally competent solutions that arise naturally

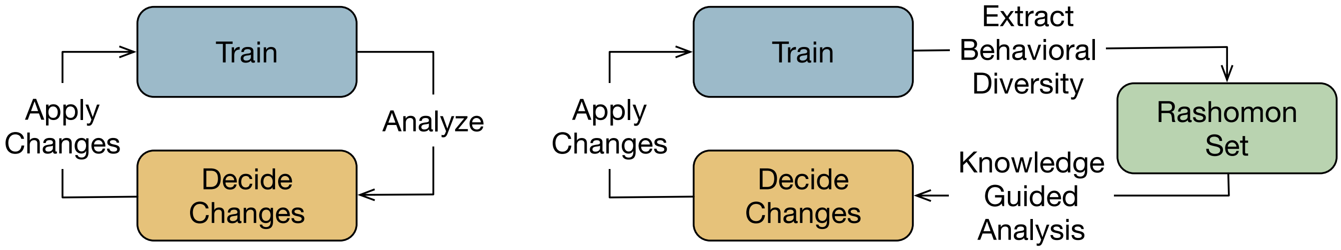

Figure 4: Traditional retraining loop (top) versus Rashomon-informed process (bottom). The Rashomon framework adds an analysis stage where behavioral diversity is examined before making changes, shifting retraining from intuition to evidence.

from stochastic initialization, random exploration, or environmental noise, even when no mechanism explicitly enforces diversity (Panda, Srivastava, and Dodge 2024).

Interpretability and Explainable RL

A parallel line of research focuses on understanding why RL agents behave as they do, emphasizing interpretability and transparency. Some approaches extract symbolic or rule-based representations from policies—such as Programmatically Interpretable RL (Verma et al. 2018) and VIPER (Bastani, Pu, and Solar-Lezama 2018)—while others employ attention mechanisms, saliency analysis, or causal reasoning to visualize decision processes (Greydanus et al. 2017; Madumal et al. 2019). More recent frameworks—such as reward decomposition and policy summarization (Septon et al. 2023; Amir and Amir 2018)—shift the emphasis from explaining how well an agent performs to why it succeeds, revealing the latent priorities encoded within a learned policy.

While these approaches have deepened our understanding of individual agent behaviors, they typically operate on a single learned policy whose explanations are generalizable. The Rashomon framework complements this view by asking what happens when multiple distinct policies achieve comparable returns, revealing how each policy may provide a different explanation for the same task—each policy reflecting a unique understanding of what the problem requires. Thus, interpretability in RL must extend beyond explaining one policy—it must encompass the diversity of explanations across the set of competent agents, capturing how different yet equally successful agents make sense of the same environment.

Conclusion and Future Directions

This work extends the Rashomon Effect to RL, framing it as a conceptual tool for analyzing behaviorally diverse yet comparable-performing policies. By unifying *behavioral diversity*, *explanation variability*, and *preference alignment* under the shared phenomenon of model multiplicity, we connect previously isolated threads in RL research. This perspective reframes the goal of analysis: understanding *how many* good policies exist and *how they differ* becomes as important as identifying a single policy that performs best. Viewing variability as an informative property of learning—rather than stochastic noise—offers a principled foundation for studying robustness, alignment, and interpretability as interdependent facets of the same underlying multiplicity.

A central open direction lies in *empirically characterizing* Rashomon sets. Future work should develop quantitative measures of behavioral diversity, trajectory divergence, and representational overlap among comparable-performing agents, as well as methods to visualize how these dimensions evolve across training regimes or reward specifications. Another important challenge is to define equivalence criteria beyond numerical return and to integrate human feedback for reasoning over *distributions* of competent policies rather than isolated models.

Understanding how many good policies exist—and why they differ—may ultimately provide a more faithful picture of intelligence than optimizing for any one of them. We therefore encourage the community to look beyond singular notions of optimality and to embrace the inherent multiplicity of RL systems.

References

Amir, D.; and Amir, O. 2018. HIGHLIGHTS: Summarizing Agent Behavior to People. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '18, 1168–1176. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems.

Amodei, D.; Olah, C.; Steinhardt, J.; Christiano, P. F.; Schulman, J.; and Mané, D. 2016. Concrete Problems in AI Safety. *ArXiv*, abs/1606.06565.

Bastani, O.; Pu, Y.; and Solar-Lezama, A. 2018. Verifiable reinforcement learning via policy extraction. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, 2499–2509. Red Hook, NY, USA: Curran Associates Inc.

Breiman, L. 2001. Statistical Modeling: The Two Cultures (with comments and a rejoinder by the author). *Statistical Science*, 16(3): 199–231.

Christiano, P. F.; Leike, J.; Brown, T. B.; Martic, M.; Legg, S.; and Amodei, D. 2017. Deep reinforcement learning from human preferences. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, 4302–4310. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781510860964.

Deb, K.; and Kalyanmoy, D. 2001. *Multi-Objective Optimization Using Evolutionary Algorithms*. USA: John Wiley & Sons, Inc. ISBN 047187339X.

Dong, J.; Rudin, C.; and Seltzer, M. 2020. Exploring the Rashomon set of sparse decision trees. In *NeurIPS*.

Eysenbach, B.; Gupta, A.; Ibarz, J.; and Levine, S. 2019. Diversity is All You Need: Learning Skills without a Reward Function. In *International Conference on Learning Representations*.

Fisher, A.; Rudin, C.; and Dominici, F. 2019. All Models are Wrong, but Many are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously. *Journal of Machine Learning Research*, 20(177): 1–81.

Frans, K.; Ho, J.; Chen, X.; Abbeel, P.; and Schulman, J. 2017. Meta Learning Shared Hierarchies. *ArXiv*, abs/1710.09767.

Greydanus, S.; Koul, A.; Dodge, J.; and Fern, A. 2017. Visualizing and Understanding Atari Agents. *ArXiv*, abs/1711.00138.

Jeffares, A.; Liu, T.; Crabbé, J.; and van der Schaar, M. 2023. Joint training of deep ensembles fails due to learner collusion. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23. Red Hook, NY, USA: Curran Associates Inc.

Kurosawa, A. 1950. Rashōmon. Film. Distributed by Daiei Film. Available at https://www.imdb.com/title/tt0042876/.

Lin, Z.; D'Oro, P.; Nikishin, E.; and Courville, A. C. 2024. The Curse of Diversity in Ensemble-Based Exploration. In *ICLR*.

Madumal, P.; Miller, T.; Sonenberg, L.; and Vetere, F. 2019. Explainable Reinforcement Learning Through a Causal Lens. In *AAAI Conference on Artificial Intelligence*.

Marx, C.; Calmon, F.; and Ustun, B. 2020. Predictive multiplicity in probabilistic classification. In *ICML*.

Ng, A. Y.; and Russell, S. J. 2000. Algorithms for Inverse Reinforcement Learning. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, 663–670. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. ISBN 1558607072.

Osband, I.; Blundell, C.; Pritzel, A.; and Roy, B. V. 2016. Deep exploration via bootstrapped DQN. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, 4033–4041. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781510838819.

Panda, S.; Srivastava, A.; and Dodge, J. 2024. Unlocking New Strategies: Intrinsic Exploration for Evolving Macro and Micro Actions. In *Intrinsically-Motivated and Open-Ended Learning Workshop @NeurIPS2024*.

Rudin, C.; Semenova, L.; and Parr, R. 2024. The Rashomon Effect of Machine Learning Models: Why We Should Care. *Nature Machine Intelligence*.

Salimans, T.; Ho, J.; Chen, X.; and Sutskever, I. 2017. Evolution Strategies as a Scalable Alternative to Reinforcement Learning. *ArXiv*, abs/1703.03864.

Semenova, L.; Rudin, C.; and Parr, R. 2022. A study in Rashomon curves and volumes: A new perspective on generalization and model simplicity in machine learning. *Journal of Machine Learning Research*, 23(69): 1–73.

Septon, Y.; Huber, T.; André, E.; and Amir, O. 2023. Integrating Policy Summaries with Reward Decomposition for Explaining Reinforcement Learning Agents. In *Advances in Practical Applications of Agents, Multi-Agent Systems, and Cognitive Mimetics. The PAAMS Collection: 21st International Conference, PAAMS 2023, Guimarães, Portugal, July 12–14, 2023, Proceedings*, 320–332. Berlin, Heidelberg: Springer-Verlag. ISBN 978-3-031-37615-3.

Van Moffaert, K.; and Nowé, A. 2014. Multi-objective reinforcement learning using sets of pareto dominating policies. *J. Mach. Learn. Res.*, 15(1): 3483–3512.

Verma, A.; Murali, V.; Singh, R.; Kohli, P.; and Chaudhuri, S. 2018. Programmatically Interpretable Reinforcement Learning. *ArXiv*, abs/1804.02477.

Vinyals, O.; Babuschkin, I.; Czarnecki, W. M.; Mathieu, M.; Dudzik, A. J.; Chung, J.; Choi, D.; Powell, R.; Ewalds, T.; Georgiev, P.; Oh, J.; Horgan, D.; Kroiss, M.; Danihelka, I.; Huang, A.; Sifre, L.; Cai, T.; Agapiou, J. P.; Jaderberg, M.; Vezhnevets, A. S.; Leblond, R.; Pohlen, T.; Dalibard, V.; Budden, D.; Sulsky, Y.; Molloy, J.; Paine, T. L.; Gulcehre, C.; Wang, Z.; Pfaff, T.; Wu, Y.; Ring, R.; Yogatama, D.; Wünsch, D.; McKinney, K.; Smith, O.; Schaul, T.; Lillicrap, T. P.; Kavukcuoglu, K.; Hassabis, D.; Apps, C.; and Silver, D. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, 575: 350 – 354.

Wood, D.; Mu, T.; Webb, A. M.; Reeve, H. W. J.; Luján, M.; and Brown, G. 2023. A unified theory of diversity in ensemble learning. *J. Mach. Learn. Res.*, 24(1).

Reproducibility Checklist

Instructions for Authors:

This document outlines key aspects for assessing reproducibility. Please provide your input by editing this `.tex` file directly.

For each question (that applies), replace the "Type your response here" text with your answer.

Example: If a question appears as

```
\question{Proofs of all novel claims
are included} {(yes/partial/no)}
Type your response here
```

you would change it to:

```
\question{Proofs of all novel claims
are included} {(yes/partial/no)}
yes
```

Please make sure to:

- Replace ONLY the "Type your response here" text and nothing else.
- Use one of the options listed for that question (e.g., **yes**, **no**, **partial**, or **NA**).
- **Not** modify any other part of the `\question` command or any other lines in this document.

You can `\input` this .tex file right before `\end{document}` of your main file or compile it as a stand-alone document. Check the instructions on your conference's website to see if you will be asked to provide this checklist with your paper or separately.

1. General Paper Structure

1.1. Includes a conceptual outline and/or pseudocode description of AI methods introduced (yes/partial/no/NA) yes

1.2. Clearly delineates statements that are opinions, hypothesis, and speculation from objective facts and results (yes/no) yes

1.3. Provides well-marked pedagogical references for less-familiar readers to gain background necessary to replicate the paper (yes/no) yes

2. Theoretical Contributions

2.1. Does this paper make theoretical contributions? (yes/no) yes

If yes, please address the following points:

2.2. All assumptions and restrictions are stated clearly and formally (yes/partial/no) yes

2.3. All novel claims are stated formally (e.g., in theorem statements) (yes/partial/no) partial

2.4. Proofs of all novel claims are included (yes/partial/no) NA

2.5. Proof sketches or intuitions are given for complex and/or novel results (yes/partial/no) yes

2.6. Appropriate citations to theoretical tools used are given (yes/partial/no) yes

2.7. All theoretical claims are demonstrated empirically to hold (yes/partial/no/NA) NA

2.8. All experimental code used to eliminate or disprove claims is included (yes/no/NA) NA

3. Dataset Usage

3.1. Does this paper rely on one or more datasets? (yes/no) no

If yes, please address the following points:

3.2. A motivation is given for why the experiments are conducted on the selected datasets (yes/partial/no/NA) NA

3.3. All novel datasets introduced in this paper are included in a data appendix (yes/partial/no/NA) NA

3.4. All novel datasets introduced in this paper will be made publicly available upon publication of the paper with a license that allows free usage for research purposes (yes/partial/no/NA) NA

3.5. All datasets drawn from the existing literature (potentially including authors' own previously published work) are accompanied by appropriate citations (yes/no/NA) NA

3.6. All datasets drawn from the existing literature (potentially including authors' own previously published work) are publicly available (yes/partial/no/NA) NA

3.7. All datasets that are not publicly available are described in detail, with explanation why publicly available alternatives are not scientifically satisficing (yes/partial/no/NA) NA

4. Computational Experiments

4.1. Does this paper include computational experiments? (yes/no) no

If yes, please address the following points:

4.2. This paper states the number and range of values tried per (hyper-) parameter during development of the paper, along with the criterion used for selecting the final parameter setting (yes/partial/no/NA) NA

4.3. Any code required for pre-processing data is included in the appendix (yes/partial/no) NA

4.4. All source code required for conducting and analyzing the experiments is included in a code appendix (yes/partial/no) NA

4.5. All source code required for conducting and analyzing the experiments will be made publicly available upon publication of the paper with a license that allows free usage for research purposes (yes/partial/no) NA

4.6. All source code implementing new methods have comments detailing the implementation, with references to the paper where each step comes from (yes/partial/no) NA

4.7. If an algorithm depends on randomness, then the method used for setting seeds is described in a way sufficient to allow replication of results (yes/partial/no/NA) NA

4.8. This paper specifies the computing infrastructure used for running experiments (hardware and software), including GPU/CPU models; amount of memory; operating system; names and versions of relevant software libraries and frameworks (yes/partial/no) NA

4.9. This paper formally describes evaluation metrics used and explains the motivation for choosing these metrics (yes/partial/no) NA

4.10. This paper states the number of algorithm runs used to compute each reported result (yes/no) NA

4.11. Analysis of experiments goes beyond single-dimensional summaries of performance (e.g., average; median) to include measures of variation, confidence, or other distributional information (yes/no) NA

4.12. The significance of any improvement or decrease in performance is judged using appropriate statistical tests (e.g., Wilcoxon signed-rank) (yes/partial/no) NA

4.13. This paper lists all final (hyper-)parameters used for each model/algorithm in the paper's experiments (yes/partial/no/NA) NA