FROM IMITATION TO UNDERSTANDING: EMERGENT EXPERT MODELLING IN SOCIAL LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent work has demonstrated that artificial agents can develop emergent social learning, using observations of expert agents to solve otherwise difficult tasks without an explicit imitation objective. In humans, such imitation is strongly linked to Theory of Mind: the ability to model the intentions, mental states, and behaviours of others. Motivated by an information-theoretic analysis of the information offered by the expert, we study whether emergent social learning analogously induces expert-modelling capabilities: the representation of the expert's latent state, goal, and behaviours. We study this question across four qualitatively different environments covering grid-worlds (GoalCycle, Overcooked), continuous control (Brax Ant), and an abstract combinatorial task (Travelling Salesman). We then test this relationship empirically and report three findings. First, social learning tracks how informative the expert is: manipulations that make the expert redundant (e.g. revealing the goal to the learner) collapse social learning and expert modelling together. Secondly, without manipulations and across independently trained agents, stronger social learning is correlated with stronger expert modelling in almost every setting. Third, the modelling view suggests concrete interventions in the form of representation-learning objectives and curiosity-style rewards that lead to large increases in both social learning and expert modelling. Together, these findings suggest a strong developmental link between social learning and modelling others.

1 INTRODUCTION

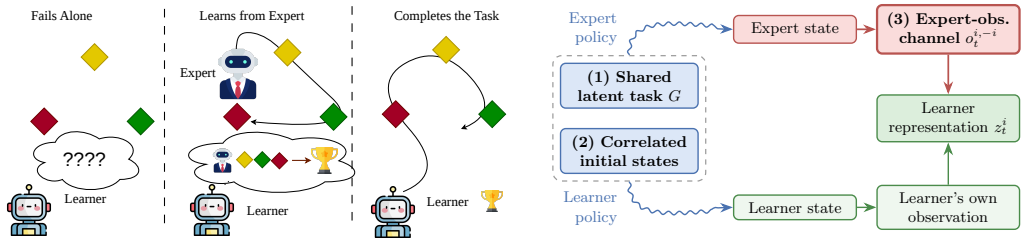
Learning from others is central to human cognitive development, and imitative social learning is closely tied to Theory of Mind: the ability to attribute mental states and model others' behaviour (Tomasello et al., 1993; Meltzoff, 1995). Recent work has shown that artificial agents can develop analogous social learning abilities. In particular, studies on emergent social learning have demonstrated that learners can benefit from the presence of expert agents—acquiring task-relevant behaviours through observation alone, without any explicit imitation objective (Ndousse et al., 2021; Bhoopchand et al., 2023; Cook et al., 2024). It remains unclear, however, whether this actually reflects any modelling of the expert.

The question of whether artificial agents can model other agents has been studied primarily as a computational problem in isolation or within cooperative settings where modelling is directly incentivised (Rabinowitz et al., 2018; Mon-Williams et al., 2025). How individual learning pressures, without any explicit social objective, can give rise to emergent expert modelling remains poorly understood. Understanding the relationship between social learning and expert modelling can help to shed light on how social learning emerges, and what can be done to encourage it.

Consider the cyclic navigation task in Figure 1a, which rewards repeatedly visiting the coloured squares in an unknown, latent order. Alone, the learner can only randomly explore; with an expert present, it can watch where the expert goes, work out the rule behind the route, and then act on it. The learner benefits from paying attention to the expert's behaviour precisely because the behaviour provides information about the hidden task, and the learner's representation, which is trained to encode reward-relevant information, should come to encode the expert's behaviour. In this case expert modelling emerges as a by-product of the learner's representation learning.

We study this effect by reimplementing and extending environments designed for studying social learning. We pair a fixed, pre-trained expert with a learner that can observe but not interact with

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.



(a) Learning from an expert (b) The learner–expert setting

Figure 1: **(a) Learning from an expert in a cyclic navigation task.** The goal is to visit coloured targets in a hidden order. Alone (left), the learner can only search by trial and error; by observing an expert traverse Yellow→Green→Red (middle), it instead infers the rule, and then uses it to complete the task (right). **(b) The learner–expert setting.** Learner (green) and expert (red) act in parallel copies of the environment, coupled by (1) a shared latent task G, (2) correlated initial states, and (3) the expert-observation channel $o_t^{i,-i}$. The expert is frozen and never observes the learner, so (3) is the only way they interact.

it (Figure 1b), across four qualitatively different environments: egocentric grid-worlds (GoalCycle, Overcooked), continuous control (Brax Ant), and an abstract optimisation task (Travelling Salesman). GoalCycle and Travelling Salesman have previously been used to study social learning (Ndousse et al., 2021; Cook et al., 2024), whereas we extend this setting to Brax Ant and Overcooked. To measure whether genuine expert modelling emerges, we develop metrics that directly capture how much the learner's internal representations encode about the expert, including the expert's actions, internal representations, subtask, and goal.

Given this framework, we test three hypotheses concerning the relationship between social learning and expert modelling that, to our knowledge, have not been systematically examined. **(H1)** Social learning and expert modelling should track the informational value of observing the expert. Manipulations that reduce this value by e.g. revealing the latent goal directly should therefore reduce both; we find this across environments. **(H2)** Across independently trained learners, stronger social learning should broadly coincide seed-for-seed with stronger expert modelling; in 3 of the 4 environments we find strong evidence of this correlation, even after taking multiple controls. **(H3)** If expert modelling is a mechanism underlying social learning, interventions that strengthen it, such as representation modelling objectives or intrinsic rewards promoting information seeking, should strengthen social learning in turn; they generally do, although the benefit of representation objectives varies across environments. In contrast to prior approaches that rely on specialised modules explicitly trained to predict other agents' actions (Rabinowitz et al., 2018; Nguyen & Gonzalez, 2020; Nguyen et al., 2023b), our results suggest that representations encoding the behaviour and goals of others can emerge as a natural consequence of learning in the presence of useful experts.

Our novel contributions are: **1.** an information-theoretic analysis showing that expert dependency, a necessary representation for social learning, is lower-bounded by partner-modelling terms; **2.** blind-corrected probe metrics that decouple expert modelling from shared environment regularities; **3.** evidence for H1–H3, relating social learning to behavioural, latent, and task modelling, and showing that representation objectives, surprisal-based intrinsic rewards, and a more legible view of the expert can strengthen social learning where it is otherwise weak.

2 Related Work

Observational learning of behaviour and goals. A long line of work studies how agents can learn from observing others. Imitation-learning approaches train models to reproduce demonstrator behaviour, either at the action level via behavioural cloning (Ross et al., 2011) or from state trajectories alone (Torabi et al., 2018). A complementary thread infers not behaviour but the underlying 'goal' of the expert: inverse reinforcement learning recovers the reward function that best explains observed behaviour (Ng et al., 2000; Ho & Ermon, 2016). Indeed, Jara-Ettinger (2019) argue that Theory of Mind can be understood through the lens of this inverse reinforcement learning. Bayesian models of human mental-state inference share this view, casting goal attribution as inverse planning over a generative model of rational agents (Baker et al., 2009). A key difference is that these methods require changing the objective of the learner towards performing this observational learning. This

means that we cannot use them to understand how agents pursuing their own objectives develop expert-modelling and social learning abilities.

Machine Theory of Mind. A complementary line of work has trained agents to directly predict properties of other agents. Rabinowitz et al. (2018) introduced the ToMnet architecture and showed limited evidence of false-belief reasoning, while subsequent multi-agent work has equipped agents with explicit 'Theory of Mind' modules or modelling objectives, typically in cooperative or competitive settings where modelling others is directly incentivised (Yu et al., 2022; Nguyen et al., 2023a; Foerster et al., 2018; Yuan et al., 2021; Wen et al., 2019; Papoudakis et al., 2021; Sclar et al., 2022; Nguyen & Gonzalez, 2020; Nguyen et al., 2023b). Most relevant to our work, Mon-Williams et al. (2025) investigate the emergence of partner-ability modelling within cooperative contexts, finding that partner-modelling can emerge when it is instrumentally useful for achieving high reward. In contrast, in our setting the relationship with the expert is non-interactive, so partner representations can only emerge through individualistic, imitative, and observational pressures that may better reflect the course of human development (where imitation in particular develops early (Tomasello, 2016)).

Emergent social learning. Our setting builds directly on prior work on emergent social learning. Ndousse et al. (2021) introduced *GoalCycle*, demonstrating that learners can emergently benefit from the presence of experts.Bhoopchand et al. (2023) extended this formulation to 3d environments, introduced metrics for studying social learning (cultural transmission metric); Cook et al. (2024) extended this to intergenerational transmission and to abstract task distributions, including a Travelling-Salesman formulation that we adopt here. These works frequently rely on interventions that themselves amount to inducing forms of expert modelling, for instance, an auxiliary objective tracking the expert's position (Bhoopchand et al., 2023), or a prestige signal that draws attention to the expert (Ndousse et al., 2021).We study how such mechanisms affect expert modelling and social learning in Section 7.3.These works have also largely been limited to variations of the GoalCycle environment, and attempts to scale this to open-ended environments like Multi-Agent Craftax (Ye et al., 2025) failed to show evidence of social learning. Additionally, these works do not actually analyse whether or how expert modelling occurs.We address this gap by directly studying the learner's internal representations across a broader set of environments.

3 The In-Context Social Learning Setting

3.1 Multi-task POMDPs requiring in-context learning

We model each agent's environment as a Partially Observable Markov Decision Process (POMDP) (Kaelbling et al., 1998), defined by a tuple $\langle S, A, T, R, O, \gamma \rangle$ of states, actions, transition function, reward function, observation emission function, and discount factor. The agent observes $o_t \sim O(s_t)$ rather than s_t, and must summarise its history internally. Crucially, the agent acts not on a single task but across a *distribution* of related tasks, each defined by a latent variable G sampled at the start of each episode that determines the reward function R_G (and, in some environments, the dynamics). Because G is unobserved and resampled each episode, no single policy is optimal across the distribution: the agent must infer G from within-episode experience. This can take the form of observations of its rewards on previous attempts, or from observing the behaviour of the expert.

3.2 The learner–expert setting

We place a single *learner* alongside a *fixed, pre-trained expert* that already achieves high return on the task distribution. Since the expert affects the learner only through observations, we model them as separate but correlated POMDPs, with superscripts $\cdot^i$ for the learner and $\cdot^{-i}$ for the expert. They are coupled in three ways (Figure 1b): **(1) Shared latent task**—the POMDPs typically share G, coupling rewards via $R^i = R^i_G$ and $R^{-i} = R^{-i}_G$ (we do break this to study Goal Modelling in Appendix G.5); **(2) Correlated initial states**—s^i_0, s^{-i}_0 are drawn jointly (e.g. with the same layout, thereby simulating being in the same environment but unable to directly interact); **(3) Expert-observation channel**—the learner observes the expert through $o^{i,-i}_t \sim O^{i,-i}(s^i_t, s^{-i}_t)$. This channel is one-way: the expert does not observe the learner, and its fixed policy is unaffected by the learner, making the setting individualistic rather than cooperative. With a fixed expert and one-way observation channel, the joint system is an extended POMDP for the learner, with the expert's hidden state and policy absorbed into the dynamics. We denote learner and expert representations by z^i_t and z^{-i}_t, and ask how much z^i_t depends on expert observations $o^{i,-i}_t$, which we call *expert dependency* (Section 4).

3.3 SOCIAL LEARNING AND EXPERT MODELLING

Social learning. We define *social learning* as a within-episode effect: observing the expert improves the learner's return on the current task, rather than merely changing its policy weights over training. Following Bhoopchand et al. (2023), we gate the expert-observation channel with a visibility schedule $v(t) \in \{0, 1\}$ and compare three conditions. **Solo** ($v(t) = 0$): the expert is never visible, providing a baseline of the learner's solo performance. **Full** ($v(t) = 1$): the expert is always visible; the gain over Solo is the *total* benefit of expert access, but a learner that merely tracks the expert step-by-step scores highly without inferring the task. **Halfway** ($v(t) = 1$ for $t < T/2$, then 0): the expert vanishes at the midpoint, so a pure following strategy collapses; the gain over Solo therefore isolates genuine in-context acquisition and retention. We combine the total and retained gains into a single *Social Learning Metric* (SLM), standardised across environments in Section 5.1.

Expert modelling. We quantify how much the learner's representation z_t^i encodes a property of the expert (e.g. its action or latent state), *beyond* what is available without observing it, since correlated environment states alone make the expert partly predictable (Section 5.3).

4 AN INFORMATION-THEORETIC VIEW OF SOCIAL LEARNING AND EXPERT-MODELLING

We use information theory to characterise the relationship between social learning occurring, and expert-observations being represented by the learner, and in turn how this can lead to forms of expert modelling. To achieve this we use the notion of *expert dependency* (ED): the dependency of the learner's representation on the expert observation channel, i.e. the one source of interaction with the expert. We formalize this notion of dependency in terms of conditional mutual information between the learner's representation, and the expert-observation channel:

$$ED_t \triangleq I\Big(z_t^i;\, o_t^{i,-i} \,\Big|\, z_{t-1}^i,\, o_t^i\Big)\,. \tag{1}$$

Expert Dependency Occurs under Social Learning We first note that social learning amounts to an increase in returns when the learner has access to the expert observation channel. If observing the expert improves the learner's returns, the value head's objective drives the value head to predict this difference, and it can do so only as a function of its embedding z_t^i. Formally, whenever $V_t = V(z_t^i)$, the data-processing inequality gives $ED_t \geq I(V_t; o_t^{i,-i} \mid C_t)$ with $C_t = (z_{t-1}^i, o_t^i)$ (Appendix F.1). In short, Social Learning requires expert dependency, because PPO's value loss pressures the encoder to retain expert observations to the extent that they help predict future returns. Intuitively, a learner cannot benefit from expert observations and not represent them.

Decomposing Expert Dependency High ED indicates dependence on the expert, but not whether the learner encodes the expert's actions or latent state, rather than only incidental cues such as position. We distinguish these by decomposing ED_t into expert-modelling and residual-observation terms, using simple applications of the chain rule for mutual information (Appendix F.2):

Proposition 1 (Expert-dependency decomposition)**.** *If the expert's previous action a_{t-1}^{-i} and latent state z_{t-1}^{-i} can influence the learner's latent only through the expert observation (see the appendix, Figure 8), then:*

$$ED_t = \underbrace{I\big(a_{t-1}^{-i}; z_t^i \mid C_t\big)}_{\textit{action modelling}} + I\Big(z_t^i; o_t^{i,-i} \mid a_{t-1}^{-i}, C_t\Big) \;\geq\; I\big(a_{t-1}^{-i}; z_t^i \mid C_t\big)\,, \tag{2}$$

$$ED_t = \underbrace{I\big(z_{t-1}^{-i}; z_t^i \mid C_t\big)}_{\textit{latent modelling}} + I\Big(z_t^i; o_t^{i,-i} \mid z_{t-1}^{-i}, C_t\Big) \;\geq\; I\big(z_{t-1}^{-i}; z_t^i \mid C_t\big)\,, \tag{3}$$

The first term captures whether z_t^i encodes the expert's action-generation process or latent state; the second captures residual cues.

These identities simply show that the expert dependency defined above can be cast in terms of expert-modelling. In practice it is difficult to isolate the residual terms. Simply encoding the expert's successive positions may for example be necessary for modelling the actions and behaviour of the expert. Additionally, we note that these terms relate expert dependency to the expert's *prior* actions or latent varianbles since these can be observed.

Interpretation. We have shown that for social learning to occur, there must be some degree of expert-dependency, i.e. representation of the expert observation channel. It remains an empirical question the extent to which this dependency constitutes modelling per se however. Our hypothesis is that this expert dependency largely arises through a simple factor: that the behaviours of the expert reveal information about the shared latent goal, meaning that they are useful for the learner to encode. To test this we probe for the expert's *current* action, latent state, sub-goal, and goal, none of which is visible in the current frame.

5 METHODOLOGY

5.1 TRAINING AND EVALUATION

We use three phases (Algorithm 1, Appendix A). **Phase 1:** train experts, select the highest performing expert, and freeze it. **Phase 2:** train the learner with the frozen expert while uniformly sampling full, halfway, and solo visibility conditions (Section 3.3) each episode. **Phase 3:** evaluate the converged learner on fresh rollouts to compute the modelling metrics (Section 5.3).

Architecture. Each agent has a recurrent encoder mapping observations to a latent z_t, an actor $\pi : Z \to \mathcal{A}$, and a critic $V : Z \to \mathbb{R}$ (CNN-GRU for the grids, MLP-GRU for TSP and Brax; Appendix B), following PureJaxRL (Lu et al., 2022). As in RL^2 (Duan et al., 2016; Wang et al., 2016), the previous action and reward are appended to each observation so the recurrent state can accumulate task evidence.

Objective. We optimise with Proximal Policy Optimisation (Schulman et al., 2017; Huang et al., 2022) over a batch of vectorised environments. The learner objective combines the standard clipped PPO policy and value losses $\mathcal{L}_a, \mathcal{L}_v$ (Appendix A.4) with an entropy bonus $\mathcal{H}(\pi)$ and the optional encoder auxiliary losses of Section 5.2 (with coefficients $c_l = c_o = 0$ except in the H3 experiments):

$$\mathcal{L}_{\text{total}} = \underbrace{\mathcal{L}_a + c_v\mathcal{L}_v - c_e\mathcal{H}(\pi)}_{\mathcal{L}_{\text{ppo}}} + c_l\mathcal{L}_{\text{latent}} + c_o\mathcal{L}_{\text{obs}}. \tag{4}$$

Social Learning Metric. We define our social metric in terms of the gains of **Half** and **Full** conditions over the **Solo** baseline (Section 3.3), normalised by pooled within condition variance:

$$\text{SLM} = \frac{1}{2}\left(\frac{\bar{R}_{\text{full}} - \bar{R}_{\text{solo}}}{s_{\text{pooled}}} + \frac{\bar{R}_{\text{half}} - \bar{R}_{\text{solo}}}{s_{\text{pooled}}}\right). \tag{5}$$

Positive SLM indicates benefit from the expert. SLM rescales the cultural-transmission metric of Bhoopchand et al. (2023) by within-condition variability rather than expert return, making it more comparable across reward scales for the learner (Appendix E.1); because that variability also includes differences in task difficulty, and environment stochasticity, we compare SLM levels only within an environment (Appendix E.1). In our experiments, we report the mean SLM over training checkpoints because social learning can be transient: as the learner becomes competent, it may cease to benefit from the expert even if expert observations were useful earlier in training.

5.2 AUXILIARY OBJECTIVES AND INTRINSIC REWARDS

As interventions for H3 we add (i) a **latent-prediction** loss $\mathcal{L}_{\text{latent}} = \mathbb{E}_t[\|P_\theta(z_t, a_t) - \text{sg}(z_{t+1})\|^2]$ with a transition predictor P_θ and stop-gradient target; (ii) an **observation-prediction** loss $\mathcal{L}_{\text{obs}} = \mathbb{E}_t[\|\psi(z_t, a_t) - o_{t+1}\|^2]$ with a decoder ψ, the mechanism used by Ndousse et al. (2021); and (iii) a **surprisal** intrinsic reward $r_t^{\text{int}} = s \cdot \text{norm}(\|P_\theta(z_t, a_t) - z_{t+1}\|^2)$ (default $s = 0$), broadly the curiosity reward of Pathak et al. (2017) without an inverse-dynamics model.

These interventions act through different routes. Observation prediction applies direct representational pressure: the predicted observation includes the expert-observation channel, so the encoder must represent the expert. Latent prediction encourages the recurrent state to integrate and retain information over time that is predictive of the future (Ni et al., 2024), including information gathered from the expert. Surprisal instead acts on behaviour. We view the expert as a source of information about the latent task, which the learner benefits from only if it attends to and encodes it. Surprisal effectively raises the reward of states where the expert is visible and not yet well modelled, drawing the learner towards this source until it has extracted what it can predict. It rewards any poorly modelled dynamic element, not just the expert, so we expect it to improve social learning and modelling only where the expert is informative about the task. Alternative intrinsic rewards are compared in Appendix G.6.

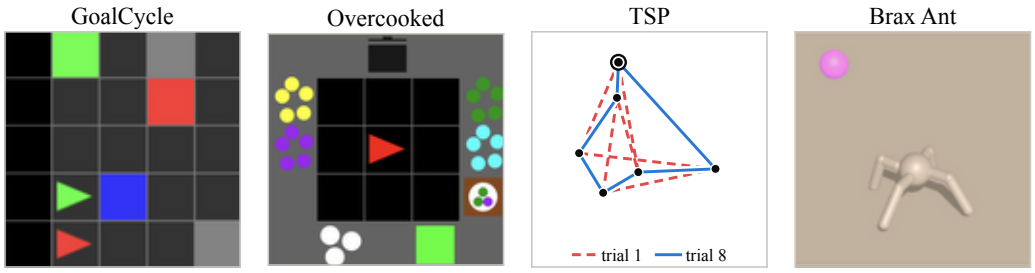


Figure 2: **The four environments** and their hidden latent G. GoalCycle and Overcooked show the expert in an egocentric view; TSP and Brax show it directly. TSP shows a random first-trial tour and a near-optimal last-trial tour of the same city layout.

5.3 Measuring expert modelling

We measure expert modelling with linear probes on the learner's representation, correcting for what is predictable without observing the expert. Since our training protocol already masks the expert channel in Solo episodes, we reuse this masking to compute *expert-blind embeddings* $z^i_{\text{blind},t}$, by rerunning the learner's encoder on the same trajectory with the expert observations masked out. For a target expert variable X_t, we train probes $q_{\text{see}}(X_t \mid z^i_t)$ and $q_{\text{blind}}(X_t \mid z^i_{\text{blind},t})$ on stop-gradient copies of the representations during Phase 2, so they do not shape the representation, and evaluate them on the set $\mathcal{D}$ of Phase-3 evaluation timesteps. With mean losses:

$$L_{\text{see}}(X) = \frac{1}{|\mathcal{D}|}\sum_{t\in\mathcal{D}} \ell\big(q_{\text{see}}(\cdot \mid z^i_t),\, X_t\big), \qquad L_{\text{blind}}(X) = \frac{1}{|\mathcal{D}|}\sum_{t\in\mathcal{D}} \ell\big(q_{\text{blind}}(\cdot \mid z^i_{\text{blind},t}),\, X_t\big), \quad (6)$$

the blind-corrected modelling metric is the fraction of the blind probe's error that observing the expert removes:

$$\Delta X\text{-Pred} = \frac{L_{\text{blind}}(X) - L_{\text{see}}(X)}{L_{\text{blind}}(X)}. \quad (7)$$

ΔX-Pred is 0 when observing the expert adds nothing beyond the rest of the learner's observations, and 1 when it removes all remaining error; since probes are evaluated on fresh rollouts it can be slightly negative, and we report it unclipped. Under certain assumptions it estimates a normalised conditional mutual information between X_t and z^i_t given $z^i_{\text{blind},t}$ (Appendix E.5).

Targets. Our main targets are the expert's next action a^{-i}_t (ΔAction-Pred), scored by the KL divergence from the expert's true policy $p(a^{-i}_t \mid s_t)$ so that its irreducible entropy is not counted as error, and its latent representation z^{-i}_t (ΔEmbedding-Pred), scored by squared error. In Overcooked we also predict the expert's next sub-task (ΔSubtask-Pred), and in misaligned GoalCycle whether the expert shares the learner's goal (ΔGoal-Pred); both are categorical and scored by cross-entropy (details in Appendix E.4).

6 Environments

In each environment, learner and expert occupy parallel, non-interacting copies sharing the same hidden latent (for full details see Appendix C). The correlated initial states are implemented as shared gridworlds (without inter-agent collision) in the case of Overcooked and GoalCycle, shared goal locations for Brax Ant, and a shared city layout for TSP (Figure 2). The observations for the gridworlds are egocentric, partial observations, whereas Brax Ant and TSP give direct access to the expert's state in the observation channel.

GoalCycle (Ndousse et al., 2021; Bhoopchand et al., 2023; Cook et al., 2024), built on XLand-MiniGrid (Nikulin et al., 2024): an $n \times n$ grid with a latent goal cycle, plus a *prestige* signal, which reveals the rewards the expert receives. **Overcooked**, a solo variant of the cooperative cooking environment (Carroll et al., 2019) built on JaxMARL Overcooked-V2 (Rutherford et al., 2024; Gessler et al., 2025): the learner must gather, cook, plate, and deliver a latent recipe; it affords a higher level target, the expert's next sub-goal (ΔSubtask-Pred). **Travelling Salesman (TSP)** (Cook et al., 2024): each episode a random six-city layout is drawn in a 1×1 square. The task is to find a minimum

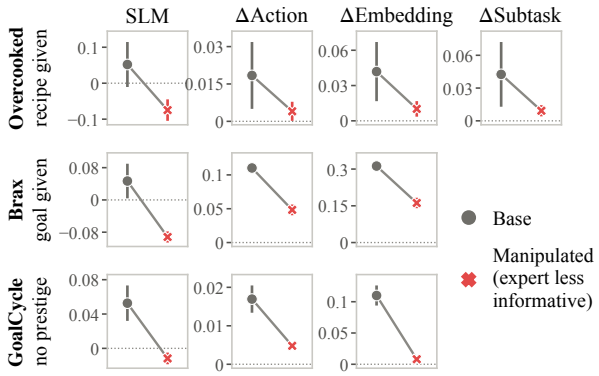


Figure 3: **Making the expert less informative collapses social learning and expert modelling together.** Base (Gray) vs. manipulated (Red). The manipulation consists in the task latent being given to the learner (Overcooked recipe, Brax goal) or prestige channel being masked (GoalCycle 7×7). Seeds per arm: 10 (Overcooked, Brax), 25 (GoalCycle).

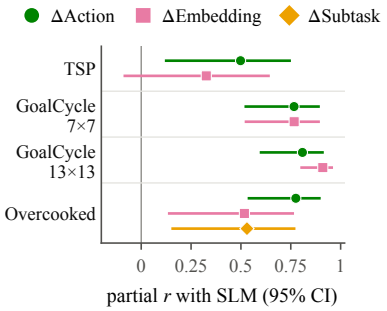


Figure 4: **SLM and expert modelling are correlated across seeds.** Partial correlation over seeds in the base condition, controlling for returns and co-presence. Confidence intervals are given by Fisher-z. Brax, where no correlation is found, is reported in Appendix G.2.

distance route between these cities in 8 trials. The learner can observe both their rewards (which provide feedback about the distances), or the experts choices in order to infer the optimal routing. **Brax Ant**, a goal-conditioned continuous control environment with the Ant embodiment (Freeman et al., 2021): the expert sees the goal; the learner sees only the expert's *goal-masked* observation and must either infer the goal from its behaviour, or use its own, dense reward signal as feedback.

7 Results

Unless otherwise noted, error bars are 95% confidence intervals over 25 seeds per condition (10 for Brax and for the Overcooked recipe-visible and expert-perspective experiments, including their matched base runs), and GoalCycle results use a penalty $p = 1.0$ for exploring in the wrong way.

7.1 H1: Social learning and expert modelling track the informativeness of the expert channel

The intuition behind Section 4 suggests that social learning and modelling should track the value-relevance of expert observations. We test this with manipulations that hold the task fixed but change how much useful information the expert channel carries.

Latent goal visibility. In Brax and Overcooked, we remove the need to infer the latent task by giving it directly to the learner (Figure 3, top two rows). In both cases social learning *and* expert modelling collapse together. SLM falls below zero (expert observations become noise), and the modelling metrics fall to about half of base in Brax and about a quarter in Overcooked, where the collapse extends to ΔSubtask-Pred.

Signalling channel (prestige). In GoalCycle, removing the *prestige signal* collapses social learning and both modelling metrics (Figure 3, bottom row). In 13×13 a surprisal reward recovers social learning in the ablated learner (Appendix G.4).

7.2 H2: Expert modelling is correlated with social learning

A key question is whether expert modelling actually co-occurs with social learning, given the same environment and task. Figure 4 reports per-seed partial correlations between SLM and each modelling metric, controlling for task returns and learner–expert co-presence, so that the relationship cannot come from some seeds simply performing better or spending more time with the expert. The relationship is strong and positive across most of the environments: the abstract combinatorial task (TSP), both sizes of GoalCycle grids, and Overcooked. This is across multiple modelling axes: the behavioural (ΔAction-Pred), latent (ΔEmbedding-Pred) and, in Overcooked, (ΔSubtask-Pred) metrics. The one weaker finding is TSP for ΔEmbedding-Pred where the interval includes zero.

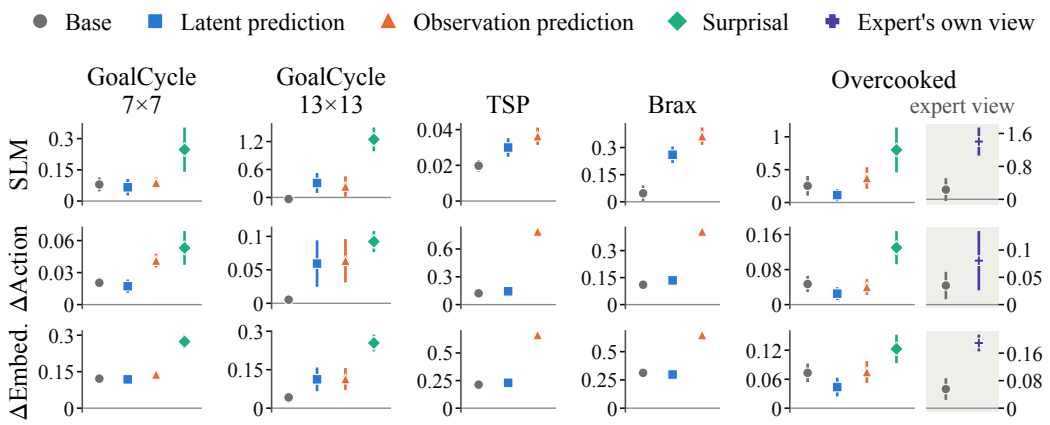


Figure 6: **Interventions that strengthen expert modelling strengthen social learning.** Latent and observation prediction are representation objectives; surprisal is an intrinsic reward, tested in GoalCycle and Overcooked. The Shaded panel ("expert view") compares the Overcooked expert-perspective variant against its own matched baseline, on its own y-scale (the variant requires masking the whole expert view, Appendix C). Seeds: 25, except Brax and the shaded panel (10).

Nevertheless, in Brax we find no evidence of a correlation (partial $r \approx -0.2$, with intervals spanning most of $-1, 1$ over 10 seeds; Appendix G.2). ΔX-Pred varies little across seeds, whereas SLM varies more substantially. Because the expert's state is directly observed at every step, expert modelling may be relatively easy and similar across learners, leaving little variation to correlate with social learning. Raw and partial correlations are reported in Appendix G.3.

Goal and plan modelling. The same seed-level relationship extends to higher-level targets (Figure 5). In the misaligned GoalCycle variant, expert and learner sample goal cycles independently, so the expert is informative only when the two happen to align. Without surprisal, neither SLM nor modelling departs far from zero. With it, about 70% of seeds show social learning, and in those seeds ΔGoal-Pred reaches 0.3–0.8 (median 0.6): the learner comes to represent whether the expert shares its task, predicting this balanced binary target at 94% accuracy, against 89% from the same representation with the expert masked and 50% in seeds that do not show social learning (Appendix G.5). In Overcooked, ΔSubtask-Pred likewise rises with SLM across seeds (partial $r = 0.53$, Figure 4), and surprisal roughly doubles it. Both targets concern what the expert is trying to do rather than its immediate behaviour, and are less immediately legible from the state.

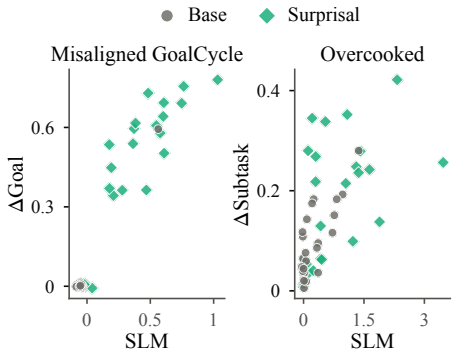


Figure 5: **Social learning correlates with modelling the expert's goal and subtask.** One point per seed, 25 per condition; SLM is the training mean. Base (grey) vs. surprisal (green). ΔGoal-Pred (misaligned GoalCycle, 13×13, $p = 1.0$) is whether the expert shares the learner's cycle; ΔSubtask-Pred is its current sub-goal.

7.3 H3: Amplifying Expert Modelling Amplifies Social Learning

Finally, we test three interventions that push expert information into the representation, expecting each to raise both modelling and social learning: making the expert more *legible*, adding a *representation objective*, and adding an information-seeking *intrinsic reward* (surprisal) (Figure 6).

Legibility. Making the expert's behaviour easier to interpret should raise both quantities without adding any new information about the task. We test this in the Overcooked *expert-perspective* variant: giving the learner the expert's own viewpoint, rather than an egocentric view of it that must be re-registered into the learner's own frame, raises SLM and ΔEmbedding-Pred well above a matched baseline, with ΔAction-Pred higher but within error (Figure 6, shaded panel).

Representation objectives. The observation-prediction objective, which forces the encoder to represent the expert as observed, raises both modelling metrics in TSP, Brax, and GoalCycle 13×13, with a corresponding rise in SLM (Figure 6). In GoalCycle 7×7 it raises only ΔAction-Pred, and SLM stays at base. Latent prediction also raises SLM in these three environments, but outside GoalCycle 13×13 its effect on modelling is smaller; in GoalCycle 7×7 it has no effect. Overcooked is the exception: observation prediction leaves SLM and modelling near baseline, while latent prediction lowers all metrics. We attribute this to task structure: the expert mainly reveals *which recipe* to cook, after which close action tracking is unnecessary, while the higher-dimensional observation may interfere with the default objective.

Surprisal. Surprisal raises SLM and both modelling metrics across the egocentric grid worlds (GoalCycle grids and Overcooked, Figure 6). The effect is particularly dramatic in the larger 13×13 grid, although this may be because its function as an expert-visibility reward is particularly helpful in navigating the larger environment. In Overcooked the two agents share a small kitchen, so the learner always has the expert in view in both conditions, yet surprisal still triples SLM, implying that it is not just functioning as a visibility-based reward however. We verify this by comparing against an expert-visibility based intrinsic reward (which underperforms surprisal; Appendix G.6) as well as ablating the particular form of surprisal used in this work.

8 Discussion

We investigated whether the kind of expert modelling associated with imitation in humans emerges as a by-product of social learning in artificial agents, without any explicit imitation or partner-modelling objective. Across four environments differing sharply in legibility and task structure, emergent social learning consistently co-occurs with representations encoding *unobserved*, agent-level properties of the expert: its current action, latent state, next sub-goal, and whether it shares the goal of the learner. The two move together across seeds (H2) and under direct manipulations of the expert's informativeness in both directions (H1, H3), so modelling is tightly coupled with social learning, and it often emerges without any dedicated objective privileging the expert (although they do work as well). The same pattern holding across diverse environments suggests the effect is not benchmark-specific, consistent with social sensitivity emerging through situated experience rather than dedicated machinery (Tomasello et al., 1993; Wellman, 2014).

Scope. Our framework applies to settings with a distribution of related tasks where the learner must perform in-context adaptation, where the context is the expert behaviour. SLM compares episodes with full, partial, and no expert visibility, and is informative only when within-episode information from the expert can change returns. Some single-task domains such as standard continuous-control benchmarks therefore do not admit a notion of social learning under this definition. Social learning is likely most beneficial when tasks are sufficiently diverse, such as in open-ended environments.

Limitations. Our environments use symbolic or state-vector observations rather than pixel-based or 3D inputs, omitting perceptual complexity that may matter for richer forms of social learning. We measure behavioural, latent, and goal modelling, but do not directly study capacities such as false-belief reasoning that are central to Theory of Mind in the cognitive-science sense. We have also not covered modelling and learning from different experts, with different levels of expertise, requiring learning a more generalisable and discriminative form of expert modelling. Our metrics are still imperfect, the blind baselines are conservative in the sense that they already include any information about the expert that is baked into the learner's trajectory.

Future work. Open-ended environments such as Craftax (Matthews et al., 2024), where social learning remains challenging (Ye & Jaques, 2024), are a natural next step. Overcooked could also compare individualistic expert modelling with partner modelling arising from cooperation (Mon-Williams et al., 2025). More generally, we would like to extend our framework to settings with two-way interactions (e.g. active teaching, or cooperative learning settings)

AI use statement

We have used AI-assisted coding tools (Github Copilot + Claude) for various tasks. Most of the architectural and training code was written by hand, but we performed lots of code refactoring and implemented subsequent methods with AI more directly. This includes editing adaptors for existing

environments, writing analysis code, and generating plots, but with human direction and verification (by the first two authors). Similarly, the text was firstly drafted by the authors, but we used AI tools for substantial editing, and have manually verified the text. This included redrafting some sections of the text, especially grouping the experiments we had run into their structure in the text. All citations were manually added, although some relevant papers were suggested by AI. Regarding required disclosure topics: we have used AI for help with developing the theoretical framework and proofs, method implementation, supporting data analysis, and feedback on the paper in general. We did not generate synthetic datasets, use translation, perform cleaning or reformatting of datasets (although we did use AI while creating adaptors/implementing RL environments.) Interpretation of results was done by the authors, although some of the interpretive text was restructured or reworded with AI assistance.

Ethics statement

This work studies representation learning in simulated agents in symbolic and physics-simulated environments. It involves no human subjects or personal data, and we do not foresee direct negative societal impacts beyond those general to reinforcement-learning research.

Reproducibility statement

The training procedure is given in Section 5.1 and Algorithm 1; hyperparameters, architectures, environment specifications, and experimental conditions are in Appendices A to D; the metric definitions and evaluation protocol are in Appendix E; and the proofs of our theoretical results are in Appendices F.1 and F.2. All environments are JAX implementations adapted from public code, and our code will be released upon publication, with scripts allowing recreation of each experiment.

References

Chris L Baker, Rebecca Saxe, and Joshua B Tenenbaum. Action understanding as inverse planning. *Cognition*, 113(3):329–349, 2009.

Avishkar Bhoopchand, Bethanie Brownfield, Adrian Collister, Agustin Dal Lago, Ashley Edwards, Richard Everett, Alexandre Fréchette, Yanko Gitahy Oliveira, Edward Hughes, Kory W Mathewson, et al. Learning few-shot imitation as cultural transmission. *Nature Communications*, 14(1):7536, 2023.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=H1lJJnR5Ym.

Micah Carroll, Rohin Shah, Mark K Ho, Tom Griffiths, Sanjit Seshia, Pieter Abbeel, and Anca Dragan. On the utility of learning about humans for human-ai coordination. *Advances in neural information processing systems*, 32, 2019.

Jonathan Cook, Chris Lu, Edward Hughes, Joel Z Leibo, and Jakob Foerster. Artificial generational intelligence: Cultural accumulation in reinforcement learning. In *Advances in neural information processing systems*, volume 37, pp. 59689–59715, 2024.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 2001. ISBN 9780471062592. doi: 10.1002/0471200611. URL https://doi.org/10.1002/0471200611.

Yan Duan, John Schulman, Xi Chen, Peter L Bartlett, Ilya Sutskever, and Pieter Abbeel. RL^2: Fast reinforcement learning via slow reinforcement learning. *arXiv preprint arXiv:1611.02779*, 2016.

Jakob N Foerster, Richard Y Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. Learning with opponent-learning awareness. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 122–130, 2018.

C. Daniel Freeman, Erik Frey, Anton Raichuk, Sertan Girgin, Igor Mordatch, and Olivier Bachem. Brax - a differentiable physics engine for large scale rigid body simulation, 2021. URL http://github.com/google/brax.

Tobias Gessler, Tin Dizdarevic, Ani Calinescu, Benjamin Ellis, Andrei Lupu, and Jakob Nicolaus Foerster. Overcookedv2: Rethinking overcooked for zero-shot coordination. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=h1vLM3GX8R.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. *Advances in neural information processing systems*, 29, 2016.

Shengyi Huang, Rousslan Fernand Julien Dossa, Chang Ye, Jeff Braga, Dipam Chakraborty, Kinal Mehta, and João G.M. Araújo. Cleanrl: High-quality single-file implementations of deep reinforcement learning algorithms. *Journal of Machine Learning Research*, 23(274):1–18, 2022. URL http://jmlr.org/papers/v23/21-1342.html.

Julian Jara-Ettinger. Theory of mind as inverse reinforcement learning. *Current Opinion in Behavioral Sciences*, 29:105–110, 2019.

Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2):99–134, 1998.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Chris Lu, Jakub Kuba, Alistair Letcher, Luke Metz, Christian Schroeder de Witt, and Jakob Foerster. Discovered policy optimisation. In *Advances in Neural Information Processing Systems*, volume 35, pp. 16455–16468, 2022.

Michael Matthews, Michael Beukman, Benjamin Ellis, Mikayel Samvelyan, Matthew Jackson, Samuel Coward, and Jakob Foerster. Craftax: A lightning-fast benchmark for open-ended reinforcement learning. In *International Conference on Machine Learning*, 2024.

Andrew N Meltzoff. Understanding the intentions of others: re-enactment of intended acts by 18-month-old children. *Developmental psychology*, 31(5):838, 1995.

Ruaridh Mon-Williams, Max Taylor-Davies, Elizabeth Mieczkowski, Natalia Velez, Neil R Bramley, Yanwei Wang, Thomas L Griffiths, and Christopher G Lucas. Partner modelling emerges in recurrent agents (but only when it matters). In *Advances in neural information processing systems*, 2025.

Kamal K Ndousse, Douglas Eck, Sergey Levine, and Natasha Jaques. Emergent social learning via multi-agent reinforcement learning. In *International conference on machine learning*, pp. 7991–8004. PMLR, 2021.

Andrew Y Ng, Stuart Russell, et al. Algorithms for inverse reinforcement learning. In *Icml*, volume 1, pp. 2, 2000.

Dung Nguyen, Hung Le, Kien Do, Svetha Venkatesh, and Truyen Tran. Social motivation for modelling other agents under partial observability in decentralised training. In *IJCAI*, pp. 4082–4090, 2023a.

Dung Nguyen, Phuoc Nguyen, Hung Le, Kien Do, Svetha Venkatesh, and Truyen Tran. Memory-augmented theory of mind network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11630–11637, 2023b.

Thuy Ngoc Nguyen and Cleotilde Gonzalez. Cognitive machine theory of mind. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 42, 2020.

Tianwei Ni, Benjamin Eysenbach, Erfan SeyedSalehi, Michel Ma, Clement Gehring, Aditya Mahajan, and Pierre-Luc Bacon. Bridging state and history representations: Understanding self-predictive rl. In *International Conference on Learning Representations*, 2024.

Alexander Nikulin, Vladislav Kurenkov, Ilya Zisman, Artem Agarkov, Viacheslav Sinii, and Sergey Kolesnikov. Xland-minigrid: Scalable meta-reinforcement learning environments in jax. In *Advances in Neural Information Processing Systems*, volume 37, pp. 43809–43835, 2024.

Georgios Papoudakis, Filippos Christianos, and Stefano Albrecht. Agent modelling under partial observability for deep reinforcement learning. volume 34, pp. 19210–19222, 2021.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pp. 2778–2787. PMLR, 2017.

Neil Rabinowitz, Frank Perbet, Francis Song, Chiyuan Zhang, SM Ali Eslami, and Matthew Botvinick. Machine theory of mind. In *International conference on machine learning*, pp. 4218–4227. PMLR, 2018.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Alexander Rutherford, Benjamin Ellis, Matteo Gallici, Jonathan Cook, Andrei Lupu, Garðar Ingvarsson, Timon Willi, Ravi Hammond, Akbir Khan, Christian Schroeder de Witt, Alexandra Souly, Saptarashmi Bandyopadhyay, Mikayel Samvelyan, Minqi Jiang, Robert Tjarko Lange, Shimon Whiteson, Bruno Lacerda, Nick Hawes, Tim Rocktäschel, Chris Lu, and Jakob Nicolaus Foerster. Jaxmarl: Multi-agent rl environments and algorithms in jax. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Melanie Sclar, Graham Neubig, and Yonatan Bisk. Symmetric machine theory of mind. In *International conference on machine learning*, pp. 19450–19466. PMLR, 2022.

Michael Tomasello. Cultural learning redux. *Child development*, 87(3):643–653, 2016.

Michael Tomasello, Ann Cale Kruger, and Hilary Horn Ratner. Cultural learning. *Behavioral and brain sciences*, 16(3):495–511, 1993.

Faraz Torabi, Garrett Warnell, and Peter Stone. Behavioral cloning from observation. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pp. 4950–4957. International Joint Conferences on Artificial Intelligence Organization, 2018.

Jane X Wang, Zeb Kurth-Nelson, Dhruva Tirumala, Hubert Soyer, Joel Z Leibo, Remi Munos, Charles Blundell, Dharshan Kumaran, and Matt Botvinick. Learning to reinforcement learn. *arXiv preprint arXiv:1611.05763*, 2016.

Henry M Wellman. *Making minds: How theory of mind develops*. Oxford University Press, 2014.

Ying Wen, Yaodong Yang, Rui Luo, Jun Wang, and Wei Pan. Probabilistic recursive reasoning for multi-agent reinforcement learning. In *International Conference on Learning Representations*, 2019.

Yilun Xu, Shengjia Zhao, Jiaming Song, Russell Stewart, and Stefano Ermon. A theory of usable information under computational constraints. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=r1eBeyHFDH`.

Eric Ye and Natasha Jaques. An efficient open world benchmark for multi-agent reinforcement learning. In *NeurIPS 2024 Workshop on Open-World Agents*, 2024.

Eric Ye, Ren Tao, and Natasha Jaques. An efficient open world environment for multi-agent social learning. *arXiv preprint arXiv:2508.15679*, 2025.

Xiaopeng Yu, Jiechuan Jiang, Wanpeng Zhang, Haobin Jiang, and Zongqing Lu. Model-based opponent modeling. *Advances in Neural Information Processing Systems*, 35:28208–28221, 2022.

Luyao Yuan, Zipeng Fu, Linqi Zhou, Kexin Yang, and Song-Chun Zhu. Emergence of theory of mind collaboration in multiagent systems. *arXiv preprint arXiv:2110.00121*, 2021.

A Training and Optimisation Details

Algorithm 1 Emergent social learning: training and evaluation. N is the number of seeds per condition (Table 1).

Phase 1 (Expert). Train N experts π^{-i} with PPO+RL^2 on the task distribution; freeze and broadcast across learner seeds
Phase 2 (Learner). Train N learner seeds alongside the frozen expert.
for each learner update **do**
  sample visibility $v \sim \{\text{full}, \text{half}, \text{solo}\}$ per parallel environment
  roll out learner π^i (batched) alongside frozen π^{-i}, gating expert observations $o_t^{i,-i}$ by $v(t)$
  compute advantages $\hat{A}_t$ and return targets $\hat{R}_t$
  for K epochs, each over M minibatches of the rollout **do**
    update θ on $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{ppo}} + c_l \mathcal{L}_{\text{latent}} + c_o \mathcal{L}_{\text{obs}}$
    update probes on $\text{sg}(z_t^i)$ to predict expert properties X^{-i} ▷ stop-gradient
  end for
  log SLM and ΔX-Pred ▷ per-seed summary = mean over training
end for
Phase 3 (Evaluation).
for each of the N learners: fresh rollouts under full/half/solo $\rightarrow$ SLM; probes on z^i vs. blind $z^i_{\text{blind},t} \rightarrow \Delta$X-Pred

A.1 Two-phase training and expert selection

In Phase 1 we train the expert alone with PPO (Section 5.1). For GoalCycle we follow Ndousse et al. (2021) in using an out-of-order-penalty curriculum so the expert reaches high-penalty competence: on the 7×7 grid we pretrain for 1000 updates at penalty $p = 0.5$ then 2000 at $p = 1.0$; on 13×13, 2000 at $p = 0.5$ then 4000 at $p = 1.0$. For the other environments the expert is trained directly for the update count in Table 1. The same number of expert seeds is trained as learner seeds, and the highest-return seed is frozen and shared across all of them within a condition, so per-seed variation reflects learner training rather than expert quality. In Phase 2 the learner trains alongside the frozen expert with visibility sampled uniformly over full/half/solo each episode. Phase 3 is the fresh-rollout evaluation of Appendix E.2.

A.2 Compute

Each run (all seeds for one condition) uses a single NVIDIA GPU—an L40 (48 GB) or A100 (80 GB), assigned by cluster availability with no methodological difference. Approximate wall-clock per run: GoalCycle $7 \times 7 \approx 1$ h, GoalCycle $13 \times 13 \approx 5$ h, Overcooked ≈ 5 h, TSP ≈ 1 h, Brax ≈ 1 h. We report 25 seeds per condition for the grid worlds, Overcooked, and TSP, and 10 for Brax and for the Overcooked latent-visible and expert-perspective variants.

A.3 Hyperparameters across environments/ experiments

We largely used default parameters for PPO (from PureJaxRLLu et al. (2022)).All environments share $\gamma = 0.99$, GAE $\lambda = 0.95$, Adam (Kingma & Ba, 2015), and advantage normalisation. For other hyperparameters, we utilised some manual tuning to train a performant expert (and then left these hyperparameters unchanged in experiments with the student). All such hyperparameters are shared between the expert and learner; final values are in Table 1. The auxiliary-loss coefficients are zero in the base condition; the latent-prediction loss ($c_l = 1.0$, also required to train the predictor used by the surprisal reward), the observation-prediction loss (c_o), and the surprisal reward ($s = 0.01$ in GoalCycle, $s = 0.1$ in Overcooked) are enabled only in the corresponding H3 conditions (Appendix D). For surprisal, we use stop gradients on the embeddings before giving them to the predictors, to ensure that they do not affect representation learning, and to isolate the effect of the surprisal signal itself.

Table 1: Per-environment training and PPO hyperparameters. GC = GoalCycle. †Total over the GoalCycle expert curriculum.

Parameter	GC 7×7	GC 13×13	Overcooked	TSP	Brax
Parallel environments	128	128	256	64	512
Rollout length	128	128	128	160	32
Max episode steps	1000	1000	400	40	512
Learner updates	1000	5000	1000	1000	1000
Expert updates	3000†	6000†	1000	1000	1000
Learning rate	1×10^{-4}	1×10^{-4}	4.3×10^{-4}	2.5×10^{-4}	2.5×10^{-4}
Gradient clip	0.5	0.5	2.1	0.5	0.5
Clip ϵ	0.11	0.11	0.19	0.2	0.2
PPO epochs	3	3	2	4	4
Minibatches	4	4	8	4	16
Entropy coeff. c_e	0.004	0.004	0.003	0.01	0.0
Value coeff. c_v	2.57	2.57	0.01	0.5	0.5
Seeds	25	25	25	25	10

A.4 PPO LOSSES

The clipped policy and value losses are the standard PPO objectives (Schulman et al., 2017). Writing $\rho_t(\theta) = \pi_\theta(a_t \mid z_t)/\pi_{\theta_{\text{old}}}(a_t \mid z_t)$ for the probability ratio, $\hat{A}_t$ for the generalised-advantage estimate, and $\hat{R}_t$ for the return target,

$$\mathcal{L}_a = -\,\mathbb{E}_t\left[\min\left(\rho_t(\theta)\,\hat{A}_t,\ \text{clip}(\rho_t(\theta),\,1-\epsilon,\,1+\epsilon)\,\hat{A}_t\right)\right], \qquad \mathcal{L}_v = \mathbb{E}_t\left[\left(V(z_t) - \hat{R}_t\right)^2\right]. \quad (8)$$

B NETWORK ARCHITECTURES

All agents share the actor–critic–encoder template of Section 5.1: a per-environment observation encoder feeds a GRU recurrent core whose latent z_t drives an actor head, a critic head, and (for the auxiliary objectives) a next-latent predictor and an optional observation decoder. Encoders are convolutional for the symbolic grid observations (Overcooked/GoalCycle) and multilayer perceptrons for the vector observations (Brax, TSP). Dimensions are in Table 2; the grid and Overcooked encoders follow Ndousse et al. (2021) with a symbolic rather than image input. The GoalCycle encoder uses SiLU activations and the Overcooked encoder ReLU.

Table 2: Network dimensions per environment. CNN = 3×3 convolutional stack; MLP = fully connected.

Component	GoalCycle	Overcooked	TSP	Brax
Observation encoder	CNN (16/64/256)	CNN (16/32) + MLP	MLP (256)	MLP
Recurrent core (GRU)	512	128	256	512
Actor head (MLP)	$512 \to 256 \to 3$	$128 \to 128 \to 6$	$256 \to 128 \to 6$	$\to 8$ (Gaussian)
Critic head (MLP)	$512 \to 256 \to 1$	$128 \to 128 \to 1$	$256 \to 128 \to 1$	$\to 1$
Next-latent predictor	$512 \to 6 \to 512$	$128 \to 6 \to 128$	$256 \to 6 \to 256$	$256 \to 6 \to 256$
Action embedding	16	16	16	—

Probes are single linear layers (with bias) applied to a stop-gradient copy of z_t^i; output dimensions match the target (action: $|\mathcal{A}|$; embedding: the expert's latent dimension; goal: 2; Overcooked sub-task: 4).

C ENVIRONMENT DETAILS

Each environment is a meta-RL task distribution with a latent variable hidden from the learner (Section 3.1). The learner and expert occupy parallel non-interacting copies sharing the same latent; the expert never observes the learner. In every environment the previous action and reward are

appended to the observation (RL^2 conditioning, Section 5.1); below we describe the remaining observation structure and the expert-observation channel, since the masking of that channel defines both the Solo condition and the blind baseline (Appendix E.3).

GoalCycle. An $n \times n$ grid (7×7, 13×13) with k goal tiles and a latent cyclic ordering sampled per episode; the next correct goal gives $+1$ and an out-of-order goal a penalty $p \in \{0, 0.5, 1.0, 1.5\}$. The observation is a $7 \times 7 \times 26$ egocentric symbolic tensor: 13 tile-type channels, 12 colour channels, and 1 *prestige* channel. Prestige is a sigmoid-transformed decaying reward statistic, $c_t = \alpha c_{t-1} + r_t$ with $\tilde{c}_t = \sigma(c_t)$ and $\alpha = 0.95$, motivated by the role of prestige in animal social learning (Ndousse et al., 2021); unlike Ndousse et al. (2021), who encode it in the colour channels, we append it as its own channel. The expert appears as a token within the learner's egocentric view; the blind baseline zeros the expert's tile/colour/prestige channels. Actions are turn-left, turn-right, and move-forward. Variants: prestige-ablated (drop the prestige channel) and misaligned (expert and learner sample independent cycles).

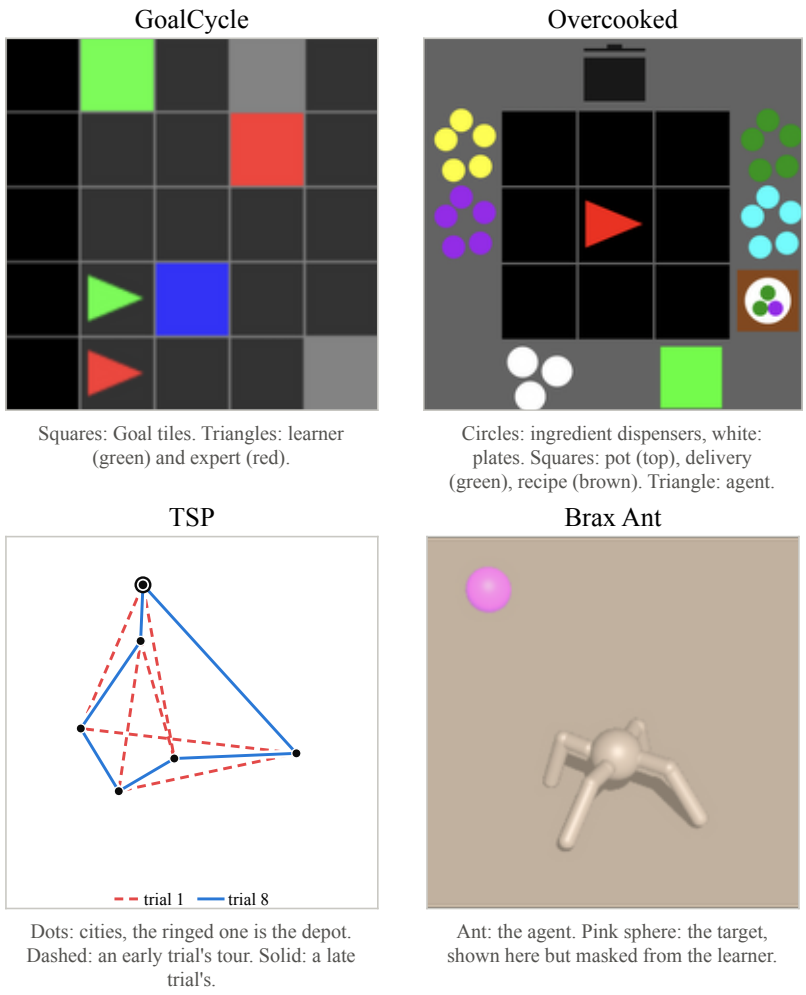
Overcooked. A solo variant of Overcooked-V2 (Carroll et al., 2019; Gessler et al., 2025) on a custom 5×5 single-room kitchen: one pot, one plate pile, one delivery counter, a recipe indicator, and four ingredient dispensers arranged around a single agent tile. Each episode samples one of the $\binom{4+3-1}{3} = 20$ unordered 3-ingredient recipes drawable from the four ingredients as the latent G. The observation is a $5 \times 5 \times 130$ symbolic tensor: 65 channels for the learner's own egocentric kitchen view, concatenated with 65 channels for an egocentric view of the expert's kitchen taken from the position the learner would occupy in the expert's copy. Under the default masking the blind baseline zeros only the expert symbol (its position, direction and inventory). With the expert's own view this would still reveal where the expert is and what it faces, so the expert-perspective variant zeros the entire expert half; because this changes both the Solo condition and the blind baseline, that variant is compared with a base run trained and probed under the same full masking scheme (10 seeds; SLM 0.24 vs. 0.25 for the default-masking base). Actions are the four moves, stay, and interact. The dense reward is in Table 3. Variants: recipe-visible (append the recipe to the learner's observation) and expert-perspective (give the learner the expert's own egocentric view rather than one from the learner's position).

Table 3: Overcooked dense reward.

Event	Reward
Correct ingredient placed	$+3$
Incorrect ingredient placed	-3
Recipe cooked	$+5$
Plate picked up	$+3$
Finished meal picked up	$+5$
Meal delivered	$+20$

Travelling Salesman (TSP). Six cities in the unit square, depot fixed at city 0; a move $a \to b$ gives $r(a, b) = (\sqrt{2} - \mathrm{dist}(a, b))/\sqrt{2}$, revisits incur -1, and the return-to-depot distance is added on a trial's final step. A meta-episode is 8 trials of 5 steps (number of cities minus one) over one fixed layout. The 15-dimensional observation contains: the agent's own previous action as a one-hot (6), a trial-terminated flag (1), the step reward (1), the expert's previous action as a gated one-hot (6), and a visibility bit (1). The layout must therefore be inferred from reward feedback or the expert's action sequence. The blind baseline zeros the expert one-hot and visibility bit. Actions select the next city (6).

Brax Ant. A goal-conditioned Ant in an open arena (Freeman et al., 2021), with the goal resampled each trial within an RL^2-style meta-episode of up to 512 steps. The reward combines a success bonus, a healthy-pose bonus, and velocity-toward-target shaping. Per-agent observations are 31-dimensional (15 qpos + 14 qvel + 2 relative target); they are assembled into an observer–target tensor in which the learner sees the expert's observation with the goal (the last two dimensions) masked out, so the learner must infer the goal from the expert's behaviour or its own reward. Actions are 8-dimensional continuous torques. Variant: goal-visible (unmask the goal for the learner).



Squares: Goal tiles. Triangles: learner (green) and expert (red).

Circles: ingredient dispensers, white: plates. Squares: pot (top), delivery (green), recipe (brown). Triangle: agent.

Dots: cities, the ringed one is the depot. Dashed: an early trial's tour. Solid: a late trial's.

Ant: the agent. Pink sphere: the target, shown here but masked from the learner.

Figure 7: **The four environments at larger size.** Renders for GoalCycle, Overcooked and Brax Ant, and a schematic for TSP showing an early trial's tour (dashed) against a late trial's (solid) over the same six cities. Below each panel is the description.

D Experimental Conditions

The main-text results reuse the training setup of Appendices A to C, varying one factor at a time. Table 4 lists each condition and its change from the base setting.

Table 4: Experimental conditions and their change from the base setting.

Condition	Used in	Environment(s)	Change from base
Base	H2	all	frozen expert; no auxiliary losses
Observation-prediction	H3	all	add observation-decoder loss ($c_o > 0$)
Latent-prediction	H3	all	add latent-prediction loss ($c_l > 0$)
Surprisal	H1, H3	GoalCycle, Overcooked	add surprisal reward ($s = 0.01$ / 0.1)
Latent-visible	H1	Overcooked, Brax	append the task latent to the learner's observation
Expert-perspective	H3	Overcooked	learner receives the expert's own egocentric view; full masking, own base
Prestige-ablated	H1	GoalCycle	remove the prestige channel
Misaligned	H2	GoalCycle	independent latents (enables ΔGoal-Pred)

E METRICS AND EVALUATION DETAILS

E.1 POOLED STANDARD DEVIATION AND RELATIONSHIP TO CULTURAL TRANSMISSION

The pooled within-condition standard deviation used to standardise SLM (Section 5.1) is a Cohen's-d pooled variance. With per-condition variances $s^2_{\text{full}}, s^2_{\text{half}}, s^2_{\text{solo}}$ over n_a, n_b, n_c episodes,

$$s_{\text{pooled}} = \sqrt{\frac{(n_a-1)s^2_{\text{full}} + (n_b-1)s^2_{\text{half}} + (n_c-1)s^2_{\text{solo}}}{n_a+n_b+n_c-3}}. \tag{9}$$

The cultural-transmission metric of Bhoopchand et al. (2023) is $\text{CT} = (\bar{R}_{\text{full}} + \bar{R}_{\text{half}} - 2\bar{R}_{\text{solo}})/(2\bar{R}_{\text{expert}})$. Our SLM shares the same numerator but normalises by s_{pooled} rather than $\bar{R}_{\text{expert}}$, hence $\text{SLM} = \text{CT} \cdot \bar{R}_{\text{expert}}/s_{\text{pooled}}$. The two therefore agree up to a per-condition scale factor; we prefer s_{pooled} because it renders the metric more comparable across environments with very different reward scales and remains well-defined when $\bar{R}_{\text{expert}}$ is near zero, at the cost of direct comparability with prior cultural-transmission numbers.

The cost of this is that pooled standard deviation also absorbs return variance that has nothing to do with the learner. In TSP each episode draws a new random city layout, and the layout accounts for most of the variance in episode return (about 96% in a simulation calibrated to the logged per-trial returns). This inflates s_{pooled}, so TSP's SLM is under-reported, by roughly 3–5$\times$ on our estimate. The inflation is nearly the same for every seed, so it rescales SLM without reordering seeds, and removing it leaves the H2 correlations essentially unchanged (with ΔAction-Pred, $r = 0.49$ either way).

E.2 EVALUATION PROTOCOL

Phase-3 evaluation rolls out each frozen learner for (Number of Parallel Environments) * 25 episodes under each of the full/half/solo conditions with fresh RNG, yielding the per-condition returns for SLM. We experimented with fitting fresh probes after training but found no significant difference from the online probes (at additional computational cost), so our results use the online probes trained during Phase 2.

E.3 BLIND BASELINES

The blind representation $z^i_{\text{blind},t}$ is produced by re-running the learner's encoder with the expert masked from its observations; the per-environment masking is given in Appendix C, and the same masking defines the Solo condition. This is done over the whole trajectory, and we keep a seperate rnn state for the blind variant throughout training and evaluation. The correction is conservative: because the learner's trajectory is itself shaped by having observed the expert, $z^i_{\text{blind},t}$ can retain indirect expert information, so ΔX-Pred *underestimates* the expert information in z^i_t.

E.4 MODELLING TARGETS

We provide additional information on the four targets used for ΔX-Pred (Section 5.1). The expert's action and latent are always available, while the subtask and goal are only defined in Overcooked and misaligned GoalCycle, respectively.

- **Action (ΔAction-Pred, all environments).** The expert's action at t, drawn from its true policy $p(a \mid s)$. We use KL divergence as the loss, which is important for interpreting the modeling metric in information-theoretic terms (Appendix E.5).
- **Embedding (ΔEmbedding-Pred, all environments).** The expert's own encoder latent z_t^{-i} (dimension in Table 2), i.e. how well the learner's representation predicts what the expert itself is representing. The loss is mean squared error, which again has a nice interpretation (Appendix E.5).
- **Subtask (ΔSubtask-Pred, Overcooked only).** Recall that Overcooked requires a sequence of picking up ingredients, placing them in the pot, picking up a plate, and delivering the finished dish. We represent the expert's current sub-goal with a hand-coded 4-way

label: PICKUP_INGREDIENT, PICKUP_PLATE, PLACE_IN_POT, DELIVER. These are recomputed deterministically every step from the expert's inventory and the pot states in its kitchen: an empty inventory yields PICKUP_PLATE if any pot is cooking or cooked and PICKUP_INGREDIENT otherwise; a held ingredient with an idle, non-full pot available yields PLACE_IN_POT; a plate or a correctly plated dish in hand yields DELIVER; a held wrong ingredient or wrong dish resets to PICKUP_INGREDIENT. Because the label is a function of the expert's internal inventory state rather than anything directly legible from a single frame of its kitchen, predicting it requires integrating the expert's behaviour over time, and predicting what it should do next (assuming it is competent).

- **Goal (ΔGoal-Pred, misaligned GoalCycle only).** A binary indicator of whether the expert's and learner's independently sampled cyclic goal orderings currently agree on the next-correct goal, i.e. whether the expert is presently informative at all (Appendix G.5).

E.5 Information-theoretic interpretation of ΔX-Pred

For log-loss targets, if each probe attains the optimal predictor in its class, the mean losses estimate conditional entropies, $L_{\text{see}}(X) \approx H(X_t \mid z_t^i)$ and $L_{\text{blind}}(X) \approx H(X_t \mid z_{\text{blind},t}^i)$, so that

$$L_{\text{blind}}(X) - L_{\text{see}}(X) \approx I(X_t; z_t^i) - I(X_t; z_{\text{blind},t}^i). \tag{10}$$

This equals $I(X_t; z_t^i \mid z_{\text{blind},t}^i)$ whenever $z_{\text{blind},t}^i$ carries no information about X_t beyond z_t^i, which is reasonable since z_t^i is computed from a superset of its inputs, giving

$$\Delta X\text{-Pred} \approx \frac{I(X_t; z_t^i \mid z_{\text{blind},t}^i)}{H(X_t \mid z_{\text{blind},t}^i)}. \tag{11}$$

For the action target, the KL loss is the cross-entropy minus the expert's policy entropy $H(a_t^{-i} \mid s_t)$. This leaves the numerator unchanged, and since the expert acts on s_t alone, the denominator becomes $I(a_t^{-i}; s_t \mid z_{\text{blind},t}^i)$: ΔAction-Pred is the fraction of the state-available action information missing from the blind embedding that the expert channel recovers. The embedding target uses squared error rather than a log loss, so it is not itself an information estimate; under Gaussian assumptions on the residuals the two are however related (Cover & Thomas, 2001).

This conditional mutual information has the same form as the modelling terms of Proposition 1, e.g. $I(a_{t-1}^{-i}; z_t^i \mid C_t)$, except that the target is the expert's *current*, unobserved variable, and the context is the blind embedding rather than C_t. Because $z_{\text{blind},t}^i$ masks the expert over the whole trajectory, ΔX-Pred captures expert information accumulated over the episode rather than only at the current step, less any that leaks through the learner's own trajectory (Appendix E.3). Since the probes are linear, these quantities are better seen in terms of the predictive $\mathcal{V}$-information of Xu et al. (2020), with $\mathcal{V}$ the family of linear probes: the numerator of ΔX-Pred estimates $H_{\mathcal{V}}(X_t \mid z_{\text{blind},t}^i) - H_{\mathcal{V}}(X_t \mid z_t^i)$, a difference of conditional $\mathcal{V}$-entropies.

F Proofs

F.1 Expert dependency and value prediction

Proposition 2 (Value prediction lower-bounds expert dependency)**.** *If the learner's value estimate is a function of its latent, $V_t = V(z_t^i)$ (as in any recurrent actor–critic), then with $C_t = (z_{t-1}^i, o_t^i)$, $ED_t \geq I\big(V_t;\, o_t^{i,-i} \mid C_t\big)$.*

Proof. By assumption, $V_t = V(z_t^i)$ is a deterministic function of z_t^i. Conditional on any context C, the chain $o_t^{i,-i} \to z_t^i \to V_t$ holds, since V_t depends on $o_t^{i,-i}$ only through z_t^i. Taking $C = C_t = (z_{t-1}^i, o_t^i)$ and applying the data-processing inequality for conditional mutual information,

$$I\Big(z_t^i;\, o_t^{i,-i} \mid C_t\Big) \;\geq\; I\Big(V_t;\, o_t^{i,-i} \mid C_t\Big).$$

The left-hand side is ED_t by definition. □

Remark 1. The bound applies per timestep and sums over t to give $ED \geq \sum_t I(V_t; o_t^{i,-i} \mid C_t)$. The per-timestep form is the more interpretable: whenever the value head's prediction at time t depends on the current expert observation (beyond what the learner already knows), the latent representation must encode that dependence.

F.2 Proof of Proposition 1

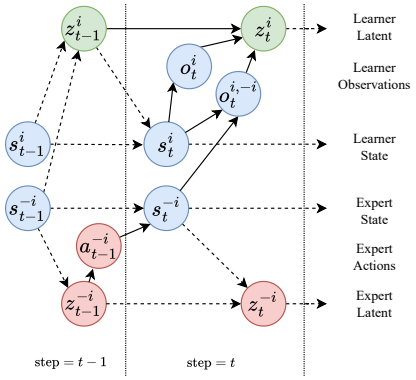


Figure 8: **Probabilistic graphical model** of the conditional relationships between the learner (green), expert (red), and environment (blue) at time t. The expert's representation z_{t-1}^{-i} and action a_{t-1}^{-i} can update the learner's latent z_t^i only through the expert observation $o_t^{i,-i}$.

Under the graphical model in Figure 8, the following Markov chains hold conditional on C_t:

$$a_{t-1}^{-i} \to o_t^{i,-i} \to z_t^i, \qquad z_{t-1}^{-i} \to o_t^{i,-i} \to z_t^i. \tag{12}$$

The two residual terms in Proposition 1 share a name but are *not* the same quantity: each is the residual of a different decomposition.

Proof. For a Markov chain $A \to B \to C$, expanding $I(A, B; C)$ by the chain rule in two ways gives

$$I(A; C) + I(B; C \mid A) \;=\; I(A, B; C) \;=\; I(B; C) + I(A; C \mid B),$$

and $I(A; C \mid B) = 0$ by conditional independence, so $I(B; C) = I(A; C) + I(B; C \mid A)$. Applying this to $a_{t-1}^{-i} \to o_t^{i,-i} \to z_t^i$ conditional on C_t (with $A = a_{t-1}^{-i}$, $B = o_t^{i,-i}$, $C = z_t^i$) gives

$$I(o_t^{i,-i}; z_t^i \mid C_t) = I(a_{t-1}^{-i}; z_t^i \mid C_t) + I(z_t^i; o_t^{i,-i} \mid a_{t-1}^{-i}, C_t),$$

whose left-hand side is ED_t, yielding the action-modelling decomposition. The same identity applied to $z_{t-1}^{-i} \to o_t^{i,-i} \to z_t^i$ yields the latent-modelling decomposition. Both lower bounds follow because conditional mutual information is non-negative. □

Our probes target the modelling terms. We do not directly estimate the residual term, since doing so would require estimating the conditional residuals above; the residual channel is assessed only indirectly, for example through the prestige ablation (Appendix G.4). It is worth noting that residual modelling (in the sense of e.g. tracking the position of the expert) may be a pre-requisite for e.g. action modelling, so these are not at odds. It is an empirical question whether the modelling terms themselves are necessary or sufficient for SLM, which we seek to address in Section 7.

G Additional Results and Analyses

G.1 Coherence of the modelling metrics

Our two modelling metrics, ΔAction-Pred and ΔEmbedding-Pred, are strongly correlated across seeds in every environment (Figure 9, $r \approx 0.80$–0.95), and in Overcooked ΔSubtask-Pred tracks ΔEmbedding-Pred almost exactly ($r = 0.98$). The metrics therefore capture a coherent underlying quantity rather than independent artefacts.

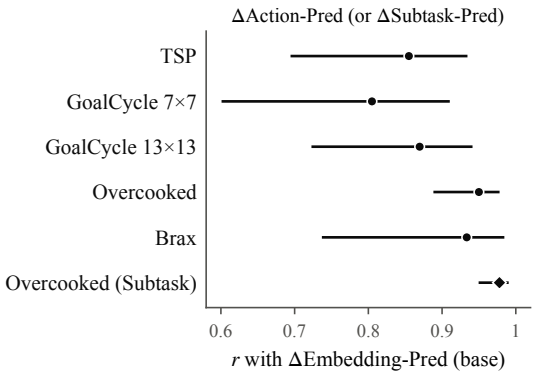


Figure 9: **Coherence of the modelling metrics.** Correlation over seeds between ΔEmbedding-Pred and ΔAction-Pred in each environment, and between ΔEmbedding-Pred and ΔSubtask-Pred in Overcooked (diamond); base condition, GoalCycle at $p = 1.0$, Fisher-z 95% CIs over 25 seeds (Brax: 10).

G.2 Absence of an SLM–modelling correlation in Brax

In Brax we find no SLM–modelling correlation: controlling for returns and co-presence, $r \approx -0.2$ for both modelling metrics over 10 seeds, with intervals spanning most of $[-1, 1]$ (Table 5). The likely reason is that ΔAction-Pred varies little across Brax seeds relative to its level (the lowest coefficient of variation of any of our environments), leaving little between-seed variation to correlate with; with 10 seeds the intervals cannot rule out a moderate correlation of either sign. The metrics are nonetheless internally consistent there (Appendix G.1). Given that the expert's state is given directly to the learner, it may be easy to model the expert incidentally without benefiting from its presence, which may also explain the weaker correlation in TSP.

G.3 Robustness of the SLM–modelling relationship

Figure 4 reports the correlation with returns and co-presence already removed. Table 5 gives the same correlations with each control applied in isolation, so the main-text value can be read against the raw correlation and against either confound alone. A final column instead removes the *other* modelling metric, isolating each metric's marginal association with SLM.

Table 5: **Partial correlations between SLM and modelling metrics.** Per-seed Pearson r in the base condition ($n = 25$; Brax $n = 10$; GoalCycle at $p = 1.0$). Columns after *raw* control for the named variable(s); $-$*Both* removes Returns and Co-presence together; $-$*partner* removes only the other modelling metric, named in the last column. "—" marks a control that is constant in that environment (co-presence is fixed in TSP). $^{***}p < .001$, $^{**}p < .01$, $^{*}p < .05$.

Env	Pair	raw r	$-$Returns	$-$Co-pres.	$-$Both	$-$partner	partner
TSP	SLM × ΔAction-Pred	$+0.49^{*}$	$+0.50^{*}$	—	—	$+0.47^{*}$	ΔEmb
	SLM × ΔEmbedding-Pred	$+0.30$	$+0.33$	—	—	-0.26	ΔAct
GC 7 × 7	SLM × ΔAction-Pred	$+0.87^{***}$	$+0.87^{***}$	$+0.77^{***}$	$+0.77^{***}$	$+0.59^{**}$	ΔEmb
	SLM × ΔEmbedding-Pred	$+0.86^{***}$	$+0.88^{***}$	$+0.72^{***}$	$+0.77^{***}$	$+0.53^{**}$	ΔAct
GC 13 × 13	SLM × ΔAction-Pred	$+0.92^{***}$	$+0.92^{***}$	$+0.88^{***}$	$+0.81^{***}$	$+0.60^{**}$	ΔEmb
	SLM × ΔEmbedding-Pred	$+0.95^{***}$	$+0.96^{***}$	$+0.95^{***}$	$+0.91^{***}$	$+0.80^{***}$	ΔAct
Overcooked	SLM × ΔAction-Pred	$+0.88^{***}$	$+0.79^{***}$	$+0.89^{***}$	$+0.78^{***}$	$+0.70^{***}$	ΔEmb
	SLM × ΔEmbedding-Pred	$+0.78^{***}$	$+0.55^{**}$	$+0.79^{***}$	$+0.52^{*}$	-0.36	ΔAct
	SLM × ΔSubtask-Pred	$+0.78^{***}$	$+0.56^{**}$	$+0.79^{***}$	$+0.53^{**}$	$+0.10$	ΔEmb
Brax	SLM × ΔAction-Pred	-0.09	$+0.12$	-0.38	-0.22	-0.05	ΔEmb
	SLM × ΔEmbedding-Pred	-0.07	$+0.21$	-0.49	-0.25	$+0.03$	ΔAct

Removing returns and co-presence leaves SLM–modelling strongly positive and significant wherever the raw correlation is significant (every environment except Brax, Appendix G.2). Where a control does reduce a correlation, the confound may be acting as a mediator: learners that seek out the expert both learn more from it and encode more of its behaviour. We treat that as suggestive rather than causal. The exception is the partner column: ΔAction-Pred and ΔEmbedding-Pred are highly collinear (Appendix G.1), so partialling one out strips much of the other's marginal association, and ΔEmbedding-Pred turns mildly negative in TSP and Overcooked. We read the two as views of one underlying quantity rather than independent predictors; Outside of Brax, SLM–ΔAction-Pred survives everywhere despite this colinearity.

Figure 10 shows how SLM and both modelling metrics move with the exploration penalty in every condition; we note the mildly surprising observation that SLM tends to be *stronger* at lower penalty.

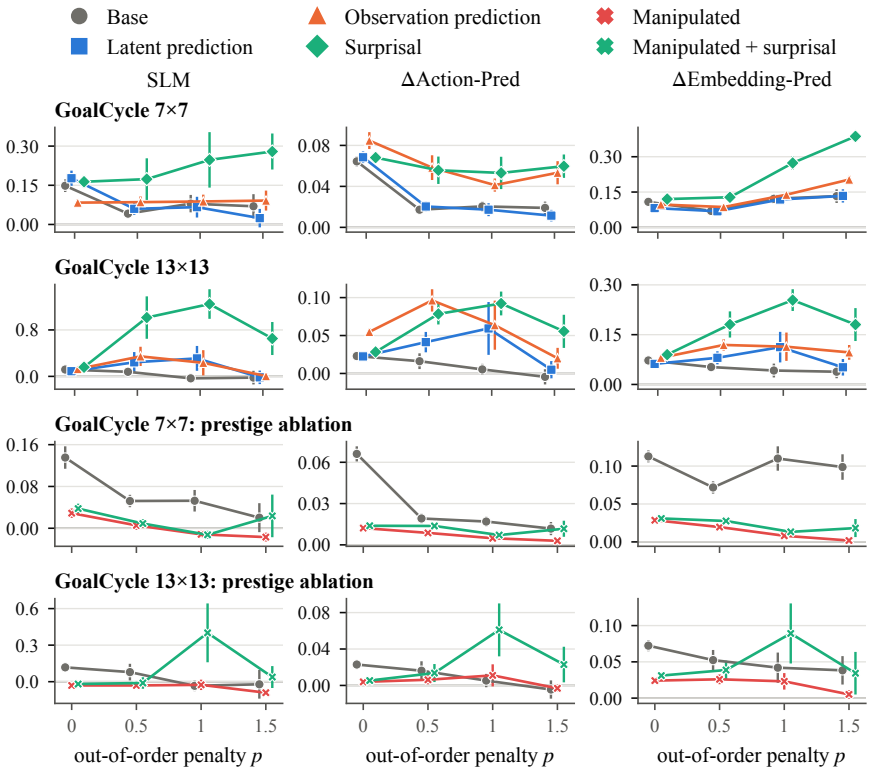


Figure 10: **Penalty sweep across GoalCycle conditions.** Rows: the 7×7 and 13×13 interventions of Section 7.3, then the prestige ablation in each grid, where "Manipulated" is the prestige-ablated variant. Columns: SLM, ΔAction-Pred, ΔEmbedding-Pred against the out-of-order penalty p. Means with 95% CIs over 25 seeds; each row has its own y-scale, so compare only within a row.

G.4 Prestige signalling in GoalCycle

The prestige signal, the sigmoid-transformed decaying reward statistic appended to observations (Appendix C), is a primary conduit of expert information in GoalCycle: removing it collapses social learning and both modelling metrics (Figure 3, bottom row, 7×7 at $p = 1.0$), and the collapse holds across penalties in both grids (Figure 10, bottom two rows). In 13×13 it is clearest at the penalties where the intact learner socially learns at all. From the perspective of Proposition 1, prestige is a powerful residual cue that directly reveals information about the reward structure, and since observing such cues is prior to behavioural and latent modelling, its absence disrupts both.

Adding the surprisal reward to the ablated learner recovers social learning, but only in 13×13 and only at the higher penalties: SLM rises from -0.026 to 0.401 at $p = 1.0$ ($p = 0.002$) and from -0.090 to 0.038 at $p = 1.5$ ($p = 0.010$), with ΔEmbedding-Pred recovering at $p = 1.0$ ($0.023 \to 0.089$, $p = 0.005$). In 7×7 no penalty shows a significant recovery (all $p > 0.19$). These are precisely

the penalties at which the *prestige-intact* 13×13 learner also fails to learn socially (SLM -0.03 at $p = 1.0$), so surprisal is not restoring the signalling channel so much as supplying an alternative route to the expert that does not depend on it.

It should be noted that in default GoalCycle the prestige channel can act as a confound for the embedding metrics, since the expert's latent is itself trained to be predictive of the expert's reward statistics. That the absence of prestige causes such a collapse in social learning and modelling points to the importance of the legibility of the expert observation channel in determining whether social learning can occur.

G.5 Goal modelling in Misaligned GoalCycle

In the misaligned GoalCycle variant, expert and learner sample cycles independently, so the expert is informative only when the two happen to align. With the base algorithm, the learner fails to learn from the expert (SLM ≈ 0) and the modelling metrics are likewise near zero. With the surprisal intrinsic reward we observe a bimodal distribution (Figure 11): mean SLM is either near zero or above 0.15, so we call a seed socially learning when its mean SLM exceeds 0.1. By that definition 18 of 25 seeds socially learn with surprisal, against 1 of 25 in the base condition, and those 18 show strong expert modelling. Most notably, ΔGoal-Pred spans 0.3–0.8 in these seeds, with a median of 0.6. The target is binary and balanced, seeds that do not socially learn sit at chance (50%), rendering the accuracies directly interpretable: in the socially learning seeds the learner's representation predicts agreement at 94%, against 89% for the same representation with the expert masked. That the blind figure is itself well above chance reflects the conservativeness of the correction (Appendix E.3): the learner's trajectory is shaped by having seen the expert, which is why ΔGoal-Pred understates the expert information it carries.

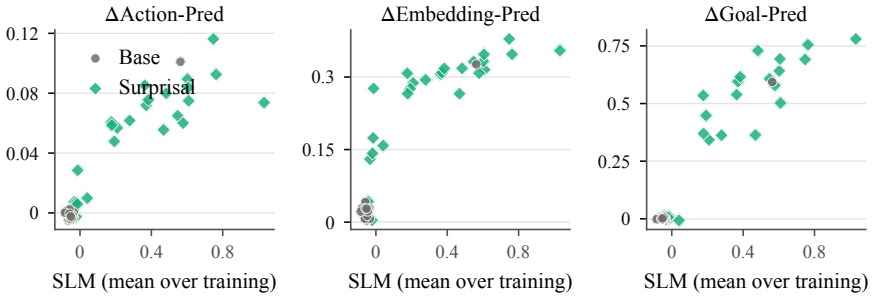


Figure 11: **Misaligned GoalCycle** (13×13, $p = 1.0$)**.** Per-seed SLM (mean over training) against ΔAction-Pred, ΔEmbedding-Pred, and ΔGoal-Pred (computed after training on fresh rollouts); 25 seeds per condition.

G.6 Alternative intrinsic rewards

Latent-prediction surprisal is motivated by the view of experts as sources of goal information, but in our simple environments the expert is also the only dynamic element outside the learner, so surprisal might simply reward proximity to the expert. There are also many other ways to improve exploration or implement curiosity-driven objectives. We compare five intrinsic rewards in both GoalCycle grids at $p = 1.0$, against the same base and surprisal runs used in Section 7.3 (Figure 12; 25 seeds per condition). All of them share the normalisation and scaling scheme $r_t^{\text{int}} = s \cdot \text{norm}(x_t)$ with $s = 0.01$, where $\text{norm}(\cdot)$ applies batch-wise z-score normalisation; the methods differ in the quantity x_t:

- **Latent-predictive surprisal (ours):** $x_t = \|\hat{z}_{t+1} - z_{t+1}\|^2$, the prediction error in the learned latent space.
- **ICM** (Pathak et al., 2017): the same latent prediction error, with the encoder additionally trained by an inverse-dynamics loss predicting a_t from (z_t, z_{t+1}). This is a variant of our reward rather than an independent alternative to it.
- **Random Network Distillation (RND)** (Burda et al., 2019): $x_t = \|\hat{f}_\theta(o_t) - f(o_t)\|^2$, where f is a fixed random target network and $\hat{f}_\theta$ is trained to match it.

- **Entropy bonus:** $x_t = \mathcal{H}(\pi(\cdot \mid z_t))$, rewarding high-entropy policies without any prediction component.
- **Expert co-presence bonus:** $x_t = \mathbf{1}[\text{expert visible in } o_t]$, a constant bonus when the expert is visible, testing whether surprisal simply acts as a proxy for staying near the expert. Unlike the others, this reward requires a privileged signal identifying the expert.

The three prediction-based rewards behave alike: surprisal, ICM and RND all raise both modelling metrics above base in both grids, and the effect is far larger in 13×13, where the base learner does not socially learn at all. Their ordering is not stable across the two grids, so we do not read the differences among them as meaningful. The entropy bonus is the exception: it fails in both, lowering ΔEmbedding-Pred in 7×7 and collapsing training outright in 13×13, so undirected stochasticity is not what drives the effect.

The co-presence control bears on the proximity worry. In 7×7 grid, there seems to be no difference. In the larger 13×13 grid, the two rewards dissociate, and the co-presence bonus appears to more significantly harm performance, and induce less social learning and modelling. Combined with the effectiveness of surprisal in the Overcooked environment where the expert is always visible, this suggests that surprisal does more than simply reward proximity. The co-presence bonus is in any case not a general method: unlike surprisal it requires a privileged signal identifying the expert, so it is unavailable wherever the expert is not labelled in advance.

Two caveats apply throughout. RND and ICM add networks and auxiliary losses that may alter training dynamics beyond the effect of the reward itself, so those two conditions should be read with more caution than the others.

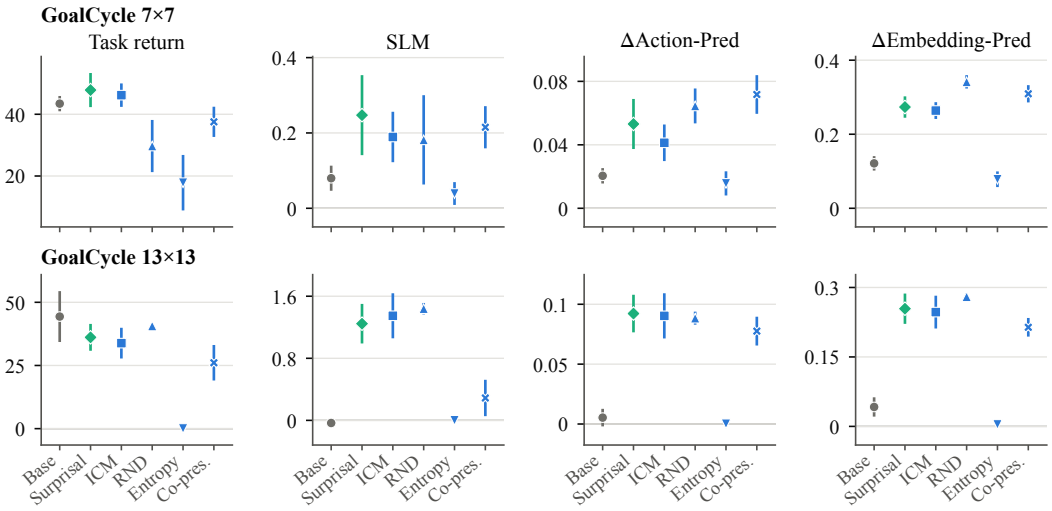


Figure 12: **Alternative intrinsic rewards** in both GoalCycle grids at $p = 1.0$. Base uses no intrinsic reward; every other condition adds one, all with the same normalisation and scale ($s = 0.01$). Means with 95% CIs over 25 seeds; within a grid, all six conditions share the same frozen expert. Each panel keeps its own y-axis—the two grids differ severalfold in SLM scale—so compare only within a row. Task return is shown alongside the metrics because several alternatives buy modelling at its expense.