Learning generalizable representations through efficient coding

Zeming Fang[1,2] and Chris Sims[2*]

[1] Brain Health Institute, National Center for Mental Disorders, Shanghai Mental Health Center, Shanghai Jiao Tong University School of Medicine and School of Psychology, Shanghai, 200030, China

[2] Department of Cognitive Science, Rensselaer Polytechnic Institute, Troy, NY, United States

* Corresponding author.

Author Note

Zeming Fang https://orcid.org/0000-0002-8091-4413

Chris Sims https://orcid.org/0000-0002-3110-1686

Abstract

Reinforcement learning (RL) theory explains human behavior as driven by the goal of maximizing reward. However, conventional RL approaches offer limited insights into human generalization. Here, we propose refining the classical RL framework by incorporating an efficient coding principle, which emphasizes maximizing reward using the simplest necessary representations. This refined framework predicts that intelligent agents, constrained by simpler representations, will inevitably: 1) distill environmental stimuli into fewer, abstract internal states, and 2) detect and utilize rewarding environmental features. Consequently, complex stimuli are mapped to compact representations, forming the foundation for generalization. We tested this model in two experiments designed to reveal human generalization abilities. Our results demonstrate that while classical RL models, which do not simplify representations, fail to generalize, an efficient coding model that learns compact representations successfully achieves human-level performance. We argue that the classical RL objective, augmented with efficient coding, represents a more comprehensive computational framework for understanding human behavior in both learning and generalization.

1. Introduction

The aphorism "A man can never step into the same river twice" speaks to the ever-changing nature of the world. Making sense of this dynamic reality requires the ability to *generalize*; that is, to extract knowledge from past experiences and apply it to new, unseen futures. Effective generalization remarkably improves the capacity of intelligent agents to adapt to rapid changes. For example, consider a child learning to ride a bike. She makes numerous attempts, falling and adjusting her balance through trial and error. Once bike riding is mastered, the child can then generalize those balancing skills to ride a scooter, allowing her to quickly master the scooter without having to learn from scratch. Given its importance to adaptive learning, generalization has been the focus of study in both cognitive neuroscience (Shepard, 1987; Shohamy & Wagner, 2008; Sims, 2018) and machine learning (Asadi et al., 2018; Li et al., 2003; Pensia et al., 2018).

Recent research illustrates that representation learning is one of the cornerstones that support generalization (Bengio et al., 2013; Goodfellow et al., 2016; Radulescu et al., 2021). Representation learning involves the transformation of raw environmental stimuli or events into robust abstract states (*"state abstraction"*), which summarize underlying patterns and regularities in the raw data. For example, a bike and scooter may be conceptually abstracted into transportation, enabling a child to realize they can transfer balancing skills previously learned from riding a bicycle to a scooter. In addition, effective representations can detect and extract a subset of the most informative and rewarding features within environments (*"rewarding feature extraction"*). For instance, although bicycles and scooters have distinct designs, their shared feature of having two wheels requires similar balancing skills. Historically, there has been a gap in the theoretical and comprehensive understanding of how to constitute effective representations. Bridging this gap and developing algorithms that learn generalizable representations has become a central pursuit in recent research on human cognitive neuroscience (Flesch et al., 2022; Nelli et al., 2023; Niv, 2019; Radulescu et al., 2021) and artificial intelligence (Bengio et al., 2013; Higgins et al., 2016; Li et al., 2006; Shwartz-Ziv, 2022; Tishby et al., 2000).

This paper focuses on understanding how humans learn effective representations that enhance their generalization abilities. One influential framework for understanding human

behavioral learning is reinforcement learning (RL), which views intelligent behavior as subservient to the maximization of reward (Silver et al., 2021; Sutton & Barto, 2018). This framework provides a normative understanding of a spectrum of human learning processes (Daw et al., 2011; Jiang et al., 2023; Niv & Langdon, 2016; Rescorla, 1972; Ribas-Fernandes et al., 2011; Tomov et al., 2021; van Opheusden et al., 2023; Xia & Collins, 2021) and offers theories on the underlying neural mechanisms (Barto, 1995; Montague et al., 1996; Niv, 2009; Schultz et al., 1997). However, by itself, the traditional RL framework provides very limited insights into human representation learning and generalization (Gershman & Daw, 2017; Ho et al., 2022; Mnih et al., 2015; Niv, 2019; Niv & Langdon, 2016). The framework often assumes a predefined, fixed set of task representations on which learning can operate directly, without the need for additional representation learning (Sutton & Barto, 2018). However, in real-world decision-making, humans are not provided with predefined representations. Instead, they must infer these representations from complex and dynamic environmental observations.

Here, we propose augmenting the classical RL theory to incorporate the principle of efficient coding (Barlow, 1961): while maximizing reward, intelligent agents should use the simplest necessary representations. Our modification to the RL framework is not unfounded because the human brain, as a biological information processing system, possesses finite cognitive resources (Miller, 1956). The idea of efficient use of cognitive resources has had profound impacts across many domains in psychology and neuroscience, including perception (Simoncelli & Olshausen, 2001; Sims, 2016; Wei & Stocker, 2015), working memory (Bates et al., 2019; Sims et al., 2012), perceptual-based generalization (Sims, 2018), and motor control (Lerch & Sims, 2021). Furthermore, our approach aligns with Botvinick's (2015) proposal that the efficient coding principle can be instrumental in understanding the representation of problems in learning and decision-making.

Critically, our proposed approach suggests that, driven by the principle of efficient coding, an intelligent agent can autonomously learn appropriate simplified representations, which enables both state abstraction and the extraction of rewarding features, naturally resulting in generalization. To validate these predictions, we designed two experiments focusing on learning and generalization. Participants first learned a set of stimulus-action associations and were then

tested on their ability to generalize to a new set of associations they had not encountered before. The first experiment investigates the emergence of state abstraction, while the second explores the extraction of rewarding features. Human participants displayed strong generalization abilities in both experiments, correctly responding to new associations without additional training. We developed a principled model based on efficient coding and demonstrated its capacity to achieve human-level generalization performance in both experiments—performance that classical RL models have not accomplished. These findings lead us to conclude that generalization is an inherent outcome of efficient coding. Given humans' remarkable capacity for generalization, we assert that the classical RL objective augmented with efficient coding and reward maximization presents a more comprehensive computational objective for human learning.

2. Results

Humans exhibit two types of generalizations: *perceptual-based* and *functional-based* generalizations. Perceptual-based generalization occurs when two stimuli share a similar appearance (Shepard, 1987; Sims, 2016; Sims et al., 2012). Functional-based generalization, in contrast, occurs between stimuli that have similar functions (*e.g.* linked to the same actions), even when they do not look alike (Collins & Frank, 2013; Collins & Frank, 2016; Meeter et al., 2009; Myers et al., 2003; Shohamy & Wagner, 2008). The latter type of generalization is more complex because it necessitates the acquisition of unseen environmental statistics before it can occur.

To investigate both types of generalization, we leveraged the *acquired equivalence* paradigm (Meeter et al., 2009; Myers et al., 2003; Shohamy & Wagner, 2008). This experimental framework first links two visually distinct stimuli with identical actions, then assesses the increase in generalization between these stimuli based on their shared actions. This approach effectively establishes the functional similarity between the two stimuli, enabling a controlled experimental investigation into participants' ability of functional-based generalization.

Specifically, participants performed a two-stage task. In each trial, participants were shown an alien (stimulus s) and were told that different aliens preferred to visit different locations. For a given stimulus, participants were required to choose one of two places (action a) that they believed the alien would prefer to visit (**Fig. 1A**). During the training stage, participants were trained on six stimulus-action associations, each repeated ten times to learn the equivalence between stimuli based on their associated actions (**Figs. 1B and 1C**). For example, if aliens s_1 and s_2 both preferred to visit a desert (a_1) rather than a forest (a_2), then they are equivalent and the psychological similarity of the two aliens may increase. During the training stage, participants received feedback (reward r, taking a value of either 0 or 1) after every choice.

In the testing stage, participants were tested on eight associations: the six trained associations plus two untrained associations that were not presented in the training stage. The untrained associations were used to evaluate people's generalization performance. For example, if the participant learned during the training stage that s_1 and s_2 were similar to each other (had similar preferences), then participants might generalize other preferences from s_1 to s_2, even

though no feedback was given about those preferences. No feedback was provided during the testing stage, and each association was repeated six times.

To quantify human generalization ability, we calculate the "untrained accuracy'', which is the response accuracy for the untrained associations that were not presented during training. The higher the untrained accuracy, the better a participant's generalization ability. Similarly, "trained accuracy"—the response accuracy for trained associations that were presented in the training stage—serves as a measure of human learning performance. Both metrics are crucial and will be used extensively throughout this paper.

All data were collected online via Amazon Mechanical Turk.

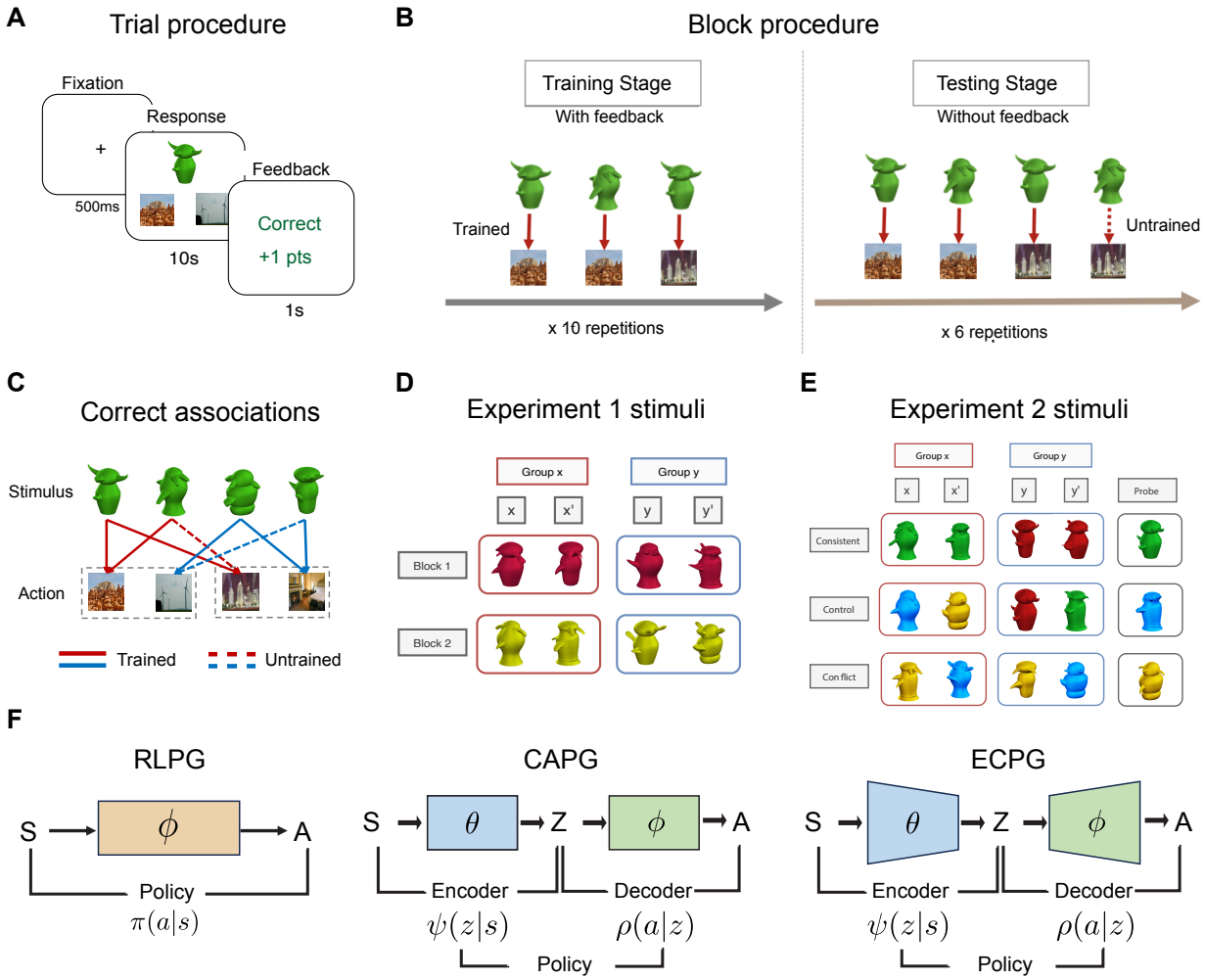

Figure 1 Acquired equivalence experimental paradigm and model architectures.

A. One trial consists of three screens: a 500-ms fixation screen, a 10-s response screen, and a 1-s feedback screen. Each response screen displays an alien stimulus, as well as two location pictures representing different actions.

B. One block contains two stages. The training stage trains three associations with feedback. The testing stage tests an untrained association (dashed line) in addition to the three trained associations without feedback.

C. One block contains two groups, each with two stimuli. The incorrect actions of one group correspond to the correct actions of the other. The structure of the stimuli is not disclosed to participants.

D. Stimuli used in Experiment 1 are designed to be the same color but with different shapes and appendages to control for perceptual similarity. The four stimuli are referred to as x, x', y, y', with x and x' associating with the same actions, as do y and y'.

E. Stimuli used in Experiment 2. Each block contains a different type of perceptual similarity.

F. The model architectures. The classical reinforcement learning policy gradient (RLPG) model learns a policy that maps from stimuli s to a distribution of action a. Due to the introduction of representation r, the policies of the cascade policy gradient (CAPG) and the efficient coding policy gradient (ECPG) model are broken into an encoder ψ and decoder ρ. The ECPG and CAPG model have the same architecture except that the ECPG model optimizes to use simpler representations z.

2.1 Modeling human behavior at the computational level

David Marr (1982) famously argued that the human brain can be understood at three levels: *the computational level,* which defines the goals to be achieved; *the algorithmic level,* which details the specific algorithms the human brain used to reach these goals; and *the implementational level*, which describes how these algorithms are physically realized.

In psychology and cognitive science, researchers often build models at the algorithmic level. They typically postulate specific cognitive mechanisms within the human brain, describe these mechanisms using computer programs, and demonstrate that incorporating them provides a better explanation of human behavioral data (*e.g.* Collins & Frank, 2013; Collins & Frank, 2016; Niv et al., 2015).

However, the question of whether the human brain reconstructs efficient representations for task stimuli is situated at the computational level. Therefore, we need to construct models at this same level. In concrete, we formalized our hypotheses—with or without efficient coding—as distinct computational goals, each addressed using the simplest possible algorithm. Crucially, unlike the algorithmic-level models, the computational-level models do not presume any specific mechanisms. Instead, these mechanisms naturally emerge during the process of achieving the defined computational goal. Thus, computational-level models not only explain human behaviors

but also shed light on the potential cognitive mechanisms underlying these behaviors, thereby demonstrating superior explanatory power over algorithmic-level models.

We built three computational-level models. First, we established a classical RL baseline, named *Reinforcement Learning Policy Gradient* (RLPG; see **Fig. 1F** and **Method 4.4.1**), which assumes that humans do not learn simplified representations. The computational goal is formulated as follows:

$$E_{\pi}[r(s_t, a_t) - b] \tag{1}$$

where $\pi(a|s)$ is a policy that maps a stimulus, s, to a distribution of actions, a. On each trial, an agent had to choose between two possible actions, each with a 50% chance of being correct. Prior to making a decision, the agent was expected to have a baseline reward expectation of b = 0.5. This baseline was used to evaluate the "goodness" of the actual reward received. A reward was considered positive if it exceeded the agent's expectation, otherwise negative. The RLPG model interpreted human behavior as involving the search for the policy that yielded the greatest reward (above the baseline) in the process of interacting with the environment.

Second, we developed an *efficient coding policy gradient* model (ECPG; **Fig. 1F** and **Method 4.4.2**), which posits that humans learn simpler representations through efficient coding. The challenge in modeling this principle lies in defining the complexity (or simplicity) of representations. Recent studies on human perception have conceptualized perception as an information transmission process, where an encoder transmits environmental sensory signals (s) into internal representation (z) (Bates et al., 2019; Sims, 2018; Sims et al., 2012). These studies measure the complexity of representations by the amount of information transmitted by the encoder, quantified by the mutual information between stimuli and representations $I^{\psi}(S; Z)$. Based on these work, the computational goal of efficient coding is formalized as maximizing reward while minimizing the representation complexity,

$$E_{\psi,\rho}[r(s_t, a_t) - b] \; - \lambda I^{\psi}(S; Z) \tag{2}$$

The critical parameter $\lambda \geq 0$, referred to as the *simplicity parameter*, controls for the tradeoff between the classical RL objective and representation simplicity. When λ = 0, the agent does not compress stimuli representations for simplicity, and the efficient coding goal reduces to the RL goal. Conversely, as $\lambda \to \infty$, the agent learns the simplest set of representations,

encoding all stimuli into a single, identical representation. Therefore, the optimal λ should be a moderate value, balancing compressing without oversimplification. Due to the introduction of latent representation z, the policy needs to be broken down into an encoder, ψ, and a decoder, ρ, which are simultaneously optimized according to Eq. 2 (**Fig. 1F**).

To test whether humans learn simpler representations, the establishment of the RLPG and ECPG models would typically be sufficient, because the contrasting hypotheses they represent (RLPG stands for "No", ECPG stands for "Yes") together cover the entire hypothesis space. One concern, however, is that the introduction of the representation in ECPG has changed the model architecture, potentially introducing confounding factors. To control these confounders, we implemented a third model, *cascade policy gradient* (CAPG; Method 4.4.3), which also supports the non-efficient coding hypothesis. The CAPG is a special case of the ECPG model which sets the simplicity parameters to 0 (λ = 0 in Eq. 2) (**Fig. 1F**),

$$E_{\psi,\rho}[r(s_t, a_t) - b] \tag{3}$$

This model serves as an intermediary between the RLPG and ECPG models, optimizing for the classical RL objective while concurrently updating the representations.

To ensure that observed behavioral differences result only from optimizing different computational goals, we carefully controlled for all other model components to avoid any confounding factors. First, all three models address their computational goals using the same *policy gradient* approach, where models explicitly learn and maintain a parameterized policy (Sutton & Barto, 2018; Williams, 1992). The method was selected over the more commonly used value function approach in psychology and neuroscience because it introduces a minimum number of parameters, therefore better distilling the computational essence of each computational goal. Second, the three models were initialized to (nearly) the same state. Due to the distinct appearances of stimuli in the experiments, we pretrained the encoders of the CAPG and ECPG models to achieve a 99% initial discrimination accuracy among the four stimuli (see **Method 4.5**). We chose a threshold of 99% instead of 100% for two reasons: first, to model the perceptual noise present in the human visual system, and second, to prevent gradient vanishing, which is an engineering concern. The RLPG model implicitly assumes perfect discrimination between stimuli and, therefore, does not require the same pretraining as others. Lastly, we used

the same model fitting method for all three models, fitting the parameters to each participant separately using maximum-a-posteriori (MAP) estimation (**Method 4.8**), based on behavioral data from both the training and the testing stages.

In the following sections, we will first demonstrate that, at the computational level, only the ECPG model—which incorporates representation simplification—can qualitatively account for human generalization behaviors (**Results 2.2** and **2.3**). We will then compare the ECPG model to several published algorithmic-level models and show that, even without presuming any specific algorithmic details about cognitive mechanisms, the ECPG model surpasses models with handcrafted cognitive mechanisms in describing human behavior (**Result 2.4**). Overall, our findings show that integrating efficient coding into the classical RL objective provides a more comprehensive computational framework for understanding human learning and generalization.

2.2 Abstract states inevitably merge in simplified representations, resulting in generalization

Experiment 1 studies human generalization using the standard acquired equivalence paradigm. In this setting, the four alien stimuli within each block are designed to have the same color but different shapes and appendages (**Fig. 1D**). This design allows us to specifically study functional-based generalization because the perceptual features (color, shape, and appendage) provide no cues for generalization.

Why can humans generalize? The proposed efficient coding principle posits that, to achieve simplified representations, an agent must appropriately abstract environmental stimuli into robust latent states. Within each of these abstract states, the stimuli can then mutually generalize. To illustrate this concept, we simulated the ECPG model at different levels of simplicity, as determined by the parameter λ (0, 0.07, 0.1, 0.2, 0.5), while keeping other parameters constant (See simulation details in **Method 4.10**). Note that when $\lambda = 0$, the ECPG model collapses to the CAPG model, without efficient coding.

In these simulations, we first demonstrate that efficient coding forces state abstraction. As shown in **Fig. 2A** ($\lambda = 0.1$), there is a significant decrease in the representation complexity of the model from the beginning of training ($t = 0$) to the end ($t = 60$). This representation simplification significantly affects the model's internal representations ($\lambda = 0.1$). Before training,

when representations are complex, each stimulus is encoded in an unstructured way, with a one-to-one correspondence in representation space (**Fig. 2D,** t = 0). Driven by efficient coding, the ECPG model compresses representations, discarding redundant information and mapping stimuli associated with the same actions into similar representations, resulting in the formation of abstract states (**Fig. 2D**, t > 20, red arrows). We quantify the degree of the state abstraction using the Silhouette score (Rousseeuw, 1987), which measures an object's (x) similarity to its own latent state (x') relative to other states (y and y'). A score close to 0 indicates that stimuli are not well-abstracted; a score close to 1 indicates that stimuli within each abstract state (x and x') are encoded similarly and associate with each other, while stimuli across abstract states (x and y) remain distinct. **Fig. 2B** (λ = 0.1) shows that the Silhouette score increases from 0, approaching 1, indicating that stable and meaningful abstract states automatically emerge from an initially unstructured set of representations. The stimuli shared the same preferences became associated with each other.

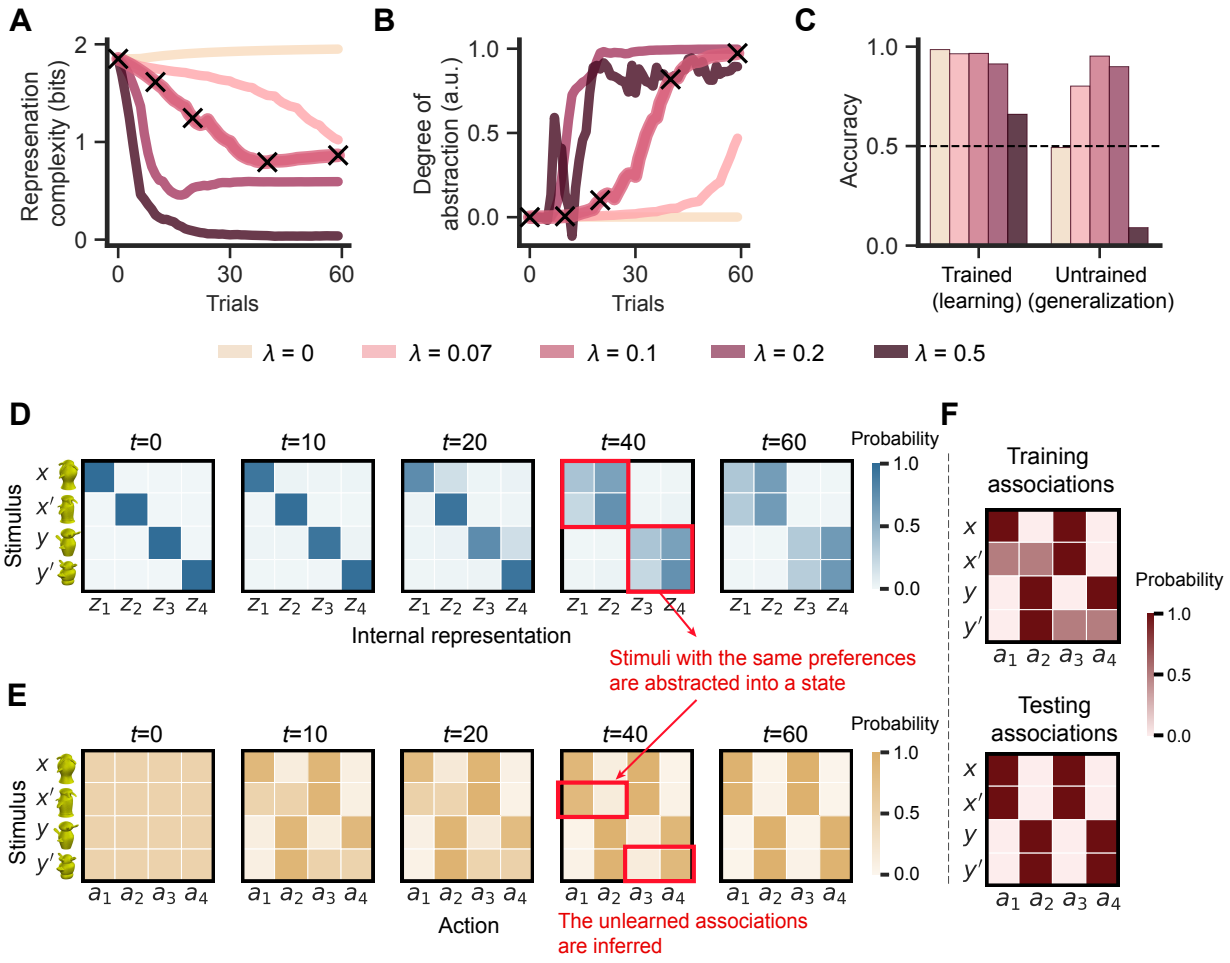

Figure 2 The internal dynamics of the ECPG model during training. Panels A-E are generated by averaging over 400 simulations. Colors in panels A-C stands for different levels of simplicity, λ.

A. Representation complexity $I^{\psi}(S;Z)$ throughout the training process. The cross makers at *t* = 0, 10, 20, 40, 60 indicate the trials that are sampled for detailed analysis.

B. Throughout the training process, the effectiveness of state abstraction is measured by the silhouette score.

C. The proportion of correct responses for associations that were presented (trained) and not presented (untrained) during the training stage reflects learning and generalization performance, respectively. This figure includes only data from the testing stage. Dashed lines represent the 50% chance level.

D. Encoders $\psi(z|s)$ with λ = 0.1 at the sampled training trials. An encoder maps a stimulus $s \in \{x, x', x, y'\}$ to a distribution of internal representations z_1 - z_4. Each row stands for a categorical distribution that sums to 1. Darker shades indicate higher probability values. See **Extended Data 5.2** for encoders for other λs.

E. Policies $\pi(a|s)$ with λ = 0.1 at the sampled training trials. A policy maps a stimulus $s \in \{x, x', x, y'\}$ to a distribution of actions a_1 - a_4. In each row, actions (a_1, a_2) and (a_3, a_4) form a probability distribution. See **Extended Data 5.2** for policies for other λs.

F. Predefined correct associations in the training and testing stages. The dark tiles indicate the correct associations, and the pink tiles stand for the associations not shown to the participants

We then illustrate that stimuli within the same abstract states can generalize to each other. After abstract states are stabilized (t > 40), the model begins to decode policy from the structured representations, and changes in representation complexity become more nuanced (**Fig. 2A**, λ = 0.1). Correspondingly, the policies decoded from stimuli within the same abstract state are also similar (**Fig. 2E,** red arrows), illustrating that the ECPG model can effectively generalize from training to testing associations. This is reflected by the model's significantly above chance-level untrained accuracy (**Fig. 2C**, λ = 0.1), despite being exposed to only a subset of the associations during training (**Fig. 2F**). Similar phenomena can be observed when λ is set to 0.2 (**Extended Data 5.3**).

Finally, we emphasize that the appropriate degree of state abstraction is critical; both insufficient and excessive abstraction impair generalization performance. As λ increases, the model shifts its focus from reward maximization to representation simplification, resulting in a more intense and rapid state abstraction, yielding simpler states (**Fig. 2A**). When λ is set to lower values (λ = 0 or 0.07), the ECPG model becomes more reward-focused and exhibits little or no reduction in representation complexity (**Fig. 2A**, λ = 0.07 and 0). The insufficient compression prevents the model from associating stimuli that share the same actions, leading to a failure in state abstraction (**Fig. 2B**, λ = 0.07 and 0) and, consequently, compromised generalization performance (**Fig. 2C**, λ = 0.07 and 0). Conversely, overly compressed representations (**Fig. 2A**, λ = 0.5) tend to oversimplify abstraction, assigning all environmental states to a single internal state. This abstraction can result in a significant reward loss, causing continuous adjustments to the abstract states, as reflected by the oscillating Silhouette score (**Fig. 2B**, λ = 0.5). Such unstable state abstraction can be detrimental to both generalization and learning performance (**Fig. 2C**, λ = 0.5).

So far, our theoretical framework has outlined how efficient coding could result in functional-based generalization. To verify whether these principles apply to the human brain, we collected a set of behavioral data where 165 participants performed two blocks of standard acquired equivalence tasks. We fitted all three models to the data and evaluated their

performance using the Bayesian Information Criterion (BIC). The ECPG model provided the best description for a majority of participants (**Fig. 3A**). The model's fitting advantage was more prominent in the testing stage, where generalization occurs, than in the learning stage (**Table 1**). Some participants' behaviors could not be accurately described by the ECPG model, primarily due to a lack of effort, which resulted in poor learning (Pearson's r = -0.96, p < 0.001) and generalization performance (Pearson's r = -0.51, p < 0.001). We also conducted a Bayesian group-level comparison and reported the protected exceedance probability (PXP)—the probability that a model accounts for the data better than the others, beyond the chance level (the log model evidence was estimated using the BIC) (Rigoux et al., 2014). As expected, the ECPG model again was ranked first among the three models (PXP > 0.999; **Fig. 1B**). These findings underscore the unique capability of the ECPG model in capturing human learning and generalization performance.

To show that only the ECPG model learns simplified representations while the two control models cannot, we computed the change in representation complexity during training. This was quantified as the difference in the mutual information $I^{\psi}(S;Z)$ before and after training (**Fig. 3C**). The ECPG model successfully reduced the complexity of its representations, indicating that it learned simplified representations as expected. In contrast, the two control models were unable to compress their representations. Furthermore, we examined the simplicity parameter λ of the ECPG model and found it to be significantly greater than 0 (t(164) = 3.493, p < 0.001, Cohen's d = 0.272) (see **Extended Data 5.1** for model parameters). This finding suggests that the representation simplification plays an important role in capturing human behaviors.

Human participants can effectively generalize in this task. Despite receiving no training, the untrained accuracy for human participants is significantly greater than the 50% chance level, though slightly lower than their accuracy for trained associations (**Fig. 3D**, blue). This observation, consistent with numerous similar studies conducted previously (Meeter et al., 2009; Myers et al., 2003; Shohamy & Wagner, 2008), indicates that human participants effectively generalized from their prior learning. The ECPG model closely captures this generalization phenomenon, whereas the two control models cannot generalize at all, with the untrained accuracy remaining at the 50% chance level (**Fig. 3D**, gray and orange). More importantly, the strong performance of the ECPG

model in capturing human generalization performance does not come at the cost of its explanatory power in human instrumental learning behavior. It offers a description as precise as that of the two control models concerning the human learning curve throughout the training stage (**Fig. 3E**).

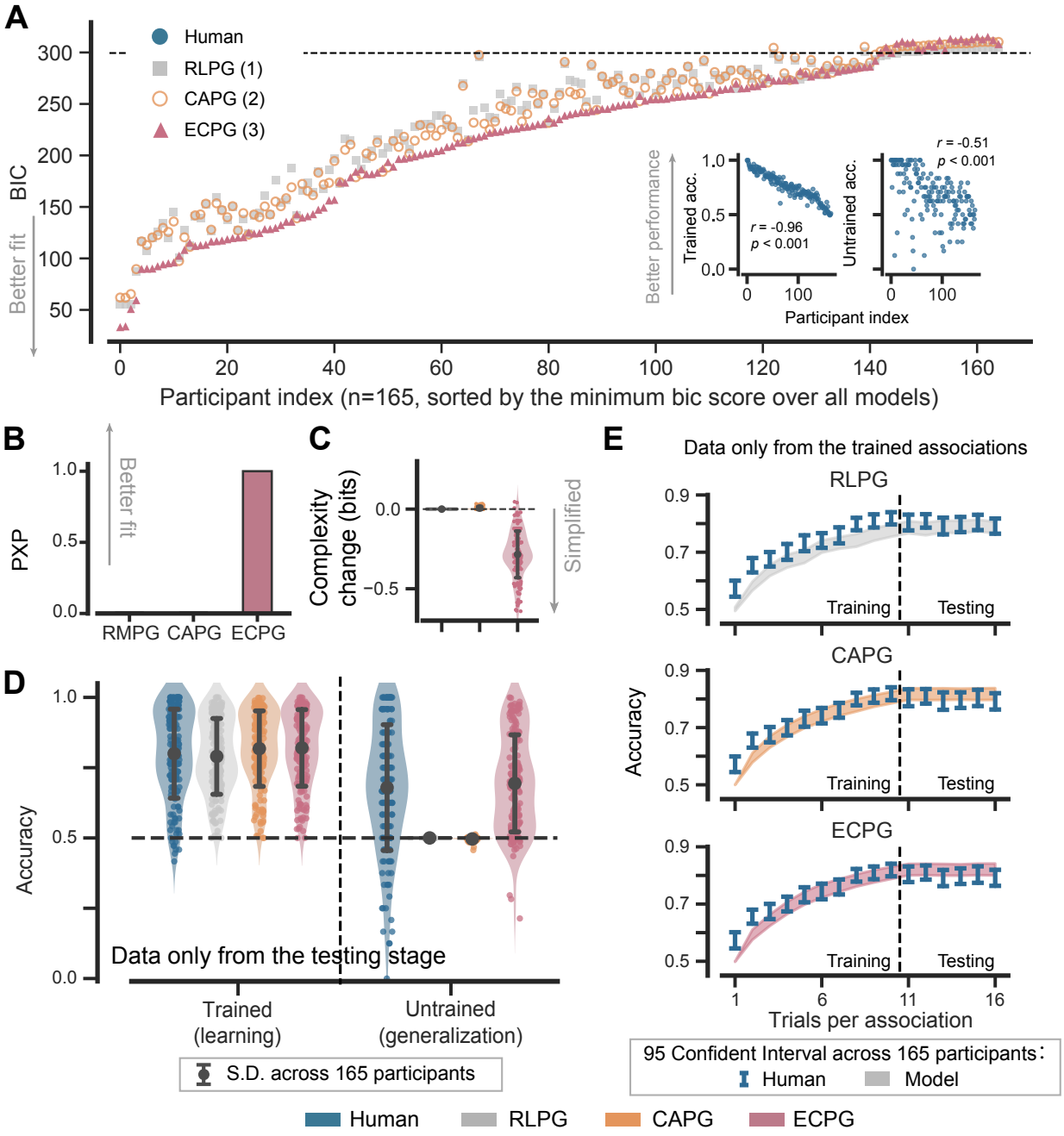

Figure 3 Behaviors of humans and models in Experiment 1. Models were fit to all behavioral data of each participant in both training and testing stages.

A. Models' Bayesian information criterion (BIC) for each participant. The RLPG model has 1 parameter, the CAPG model has 2 parameters, and the ECPG model has 3 parameters. Also, see **Table 1** for the exact value.

B. Protected exceedance probability (PXP) tallies for each model.

C. Change in representation complexity after training. Scatterplot data located above the horizontal dashed line indicate representation expansion, while those below the line indicate representation compression. The RLPG model assumes that environmental stimuli are always perfectly reconstructed and, therefore, yield no change in complexity. Error bars reflect the standard deviation across 165 valid participants.

D. Proportion of correct responses for trained and untrained associations in the testing stage, representing learning and generalization performance, respectively. Only data from the testing stage is included. The dashed lines indicate the 50% chance level. Error bars reflect the standard deviation across 165 valid participants.

E. Proportion of correct responses over the number of trials for each association. The dashed line splits the experiment into two stages: on the left is the training stage, and on the right is the testing stage. Data from untrained associations is excluded from this learning curve analysis. Error bars reflect 95% confidence interval of the mean.

Based on both quantitative and qualitative evidence, we conclude that humans' ability to generalize originates from their computational goal of efficient coding. This process promotes the emergence of abstract latent states, which form the foundational basis for generalization.

Table1. Model Fitting in Experiment 1

	Overall			**Training stage**			**Testing stage**		
	RLPG(1)	CAPG(2)	ECPG(3)	RLPG(1)	CAPG(2)	ECPG(3)	RLPG(1)	CAPG(2)	ECPG(3)
NLL	114.562	111.16	**101.202**	66.196	64.235	**62.061**	48.366	46.925	**39.141**
AIC	231.124	226.319	**208.403**	134.392	132.47	**130.121**	98.732	97.85	**84.282**
BIC	234.499	233.07	**218.529**	137.179	**138.045**	138.484	101.52	103.425	**92.644**

Bold values indicate the best for each criterion
NLL=negative log likelihood; AIC=Akaike information criterion; Bayesian information criterion
The parameter number is shown in the brackets. For example, ECPG(3) means ECPG has 3 parameters.

2.3 Efficient coding automatically extracts rewarding features throughout learning simplified representations

Experiment 2 extended the standard paradigm to examine both functional-based and perceptual-based generalizations of human participants. The experiment featured two primary modifications. First, we manipulated the stimuli's perceptual cues--shape, color, and appendage--to ensure each feature provided a different amount of information about the environment's

rewards. We designed three experimental conditions, each with a distinct rewarding configuration (**Fig. 1E**):

- In the *consistent* condition, the alien stimuli with the same color were associated with the same actions, making the color the most rewarding feature.
- In the *control* condition, the colors of the stimuli were mutually different, and all features were equally rewarding. This condition, like Experiment 1, only tested the functional-based generalization and the state abstraction ability of an agent.
- In the *conflict* condition, stimuli with the same color were associated with different actions, making shapes and appendages the rewarding features, while the color cue yielded a negative reward.

These three conditions also indicated three levels of difficulty in rewarding feature extraction. In the consistent and conflict condition, the four stimuli shared two colors, making color cues more frequent and salient. For example, while the "cylinder" shape was associated with rewards twice during training, the color "red" might have been rewarded four times. The consistent condition was the easiest because this salient feature yielded positive rewards, while the conflict condition was the most difficult because the agent needed to first suppress the color cue, the salient feature, before being able to detect rewarding ones.

The second primary modification in Experiment 2 was the incorporation of a *probe* stimulus during the testing stage; this stimulus was entirely new and had not been encountered during training. This probe was used to assess humans' ability to extract informative and rewarding features at a behavioral level. A more detailed introduction to the use of this probe design follows below, along with the presentation of our model's predictions. All other aspects of the experiment remained identical to those in Experiment 1.

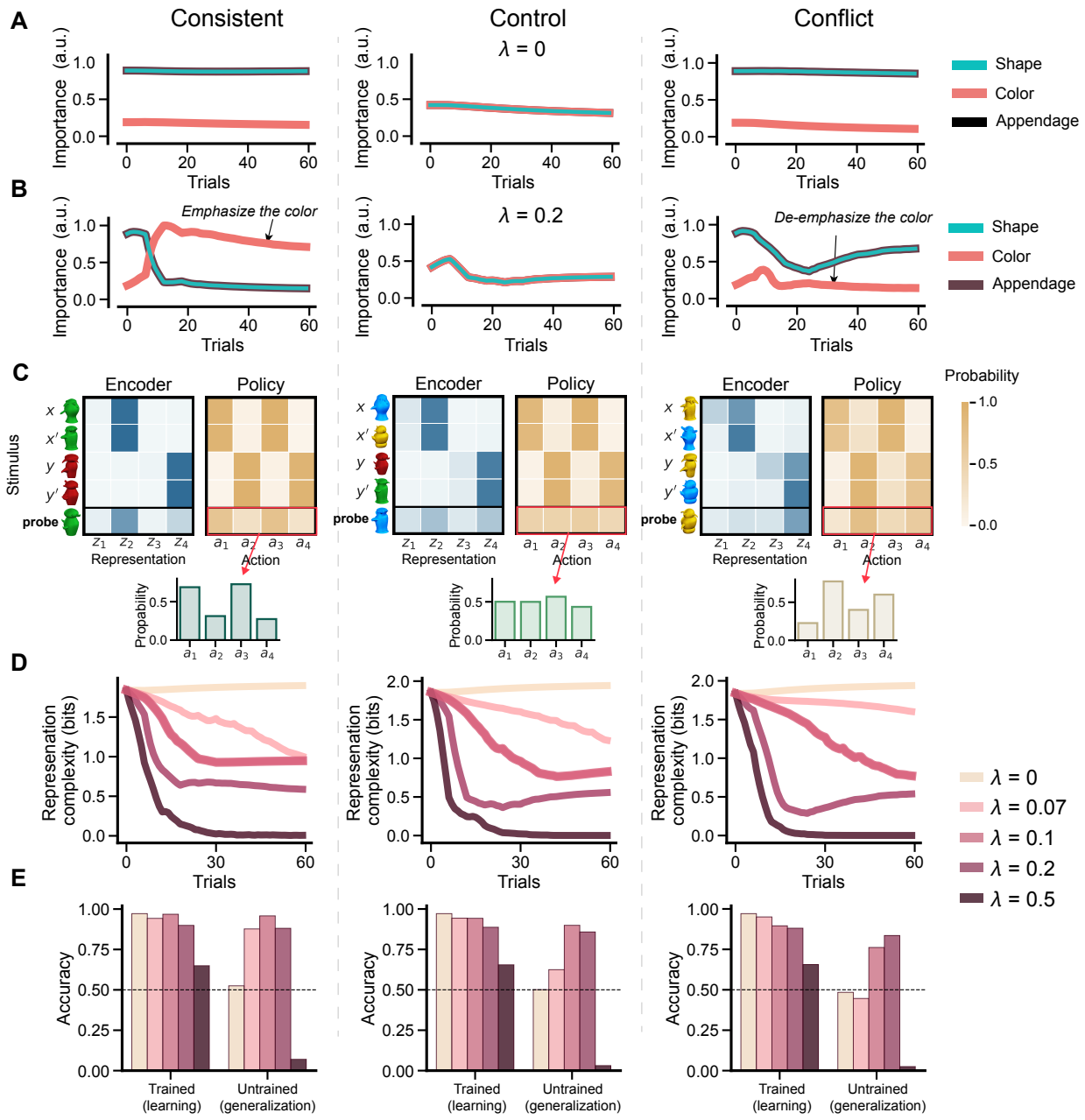

Figure 4 Predictions of the fECPG model in Experiment 2. All panels are generated by averaging over 400 simulations.

A. Simulated feature importance along with training of the fECPG model with λ = 0, which collapses to the fCAPG model. Note that the "shape" and "appendage" curves always overlap.

B. Simulated feature importance along with training of the fECPG model with λ = 0.2.

C. The predictive "probe" representation and policy of the fECPG agent (λ = 0.2) at the end of the training stage. An encoder maps a stimulus $s \in \{x, x', x, y'\}$ to a distribution of internal representations z_1 - z_4, and a policy maps a stimulus $ss \in \{x, x', x, y'\}$ to a distribution of internal representations a_1 - a_4. Darker tiles denote higher values. The policies for the control and conflict conditions are more stochastic than are those in the

consistent condition. The predictive policies applied to the probe stimuli are visualized in both a heatmap and a bar plot.

D. Representation complexity throughout learning.

E. Learning and generalization performance for the fECPG model with different levels of simplicity, λ = 0, .07, 0.1, 0.2, 0.5.

We reused the three models in Experiment 1, with only one key modification: adding a *feature embedding function* to encode the stimuli's perceptual information. The feature embedding function encoded each of the three visual features into a five-dimensional one-hot code, where each dimension indicated a specific feature value. For example, the shape "cylinder" was [1, 0, 0, 0, 0], the color "purple" was [1, 0, 0, 0, 0], and the color "yellow" was [0, 1, 0, 0, 0]. Each stimulus was represented by a combination of three such codes, concatenated into a 15-dimensional vector to form the model's input. We refer to the models used in Experiment 2 as feature RLPG (fRLPG; **Method 4.4.4**), feature CAPG (fCAPG; **Method 4.4.6**), and feature ECPG (fECPG; **Method 4.4.5**) models to highlight their integration of the feature embedding construct.

We evaluated the model's feature extraction ability by analyzing the *importance* it assigned to each feature, which quantifies each feature's contribution to the model's encoding process (Greydanus et al., 2018; Guo et al., 2021). Specifically, we added noise to one feature dimension of the stimuli to perturb their appearance and examined the corresponding effect on representations. A larger change in the representations indicated a higher importance of the perturbed feature (see **Method 4.7**). Therefore, in this experiment, if a model consistently assigns more importance to the predefined rewarding perceptual cue across all three experimental conditions, we conclude that this model can effectively detect and extract rewarding features within an environment.

Now that the stage is set, we can focus on answering three central research questions. First, can the principle of efficient coding drive a model to extract rewarding features? Second, if so, how do we validate that humans follow this principle in their learning processes? Third, how does the rewarding feature extraction interact with the state abstraction ability examined in Experiment 1?

The answer to the first question is positive: efficient coding does promote rewarding feature extraction. As designed in the experiment, color served as the rewarding feature in the consistent condition, while it yielded negative rewards in the conflict condition. Driven by the

need for simpler representations—reflected in a focus on fewer features—the fECPG model (λ = 0.2) must selectively assign more importance to the color cue in the consistent case; and less importance when the color becomes unrewarding (**Fig. 4B**, consistent). Conversely, in the conflict condition, the model had to first deemphasize the salient cue color, due to its negative rewards, and then reallocate importance to the other features contributing to positive rewards (**Fig. 4B**, conflict). The demand for simplicity drives the model to focus on a subset of features, and the goal of maximizing reward ensures that these focused features must be rewarding. In contrast, a model without efficient coding ($\lambda = 0$) cannot adaptively reallocate feature importance during its interaction with the environment. The model exhibits nearly the same feature importance assignment for both consistent and conflict cases (**Fig. 4A**), indicating its inability to detect rewarding information. It is worth noting that the fECPG model unintuitively predicts that shape and appendage are rewarding features before training. We believe this is caused by our simplistic approach to encoder initialization. We will further elaborate this point in the discussion section. However, this observation does not undermine our conclusion.

To address the second question, we adopted a "probe" design within each block to validate the theoretical predictions of the fECPG model on human participants. The probe stimulus, introduced only during the testing stage and not present in the training, was designed to always share the same color as stimulus x and the same shape as stimulus y'. According to predictions of the fECPG model, humans should employ different policies in response to the probe stimulus for different conditions. In the consistent condition, where color was the most important feature, the probe stimulus should be perceived as similar to stimulus x (**Fig. 4C**, consistent, encoder), leading to a response that coincides with the one for stimulus x (**Fig. 4C**, consistent, policy). In this scenario, it is expected that human participants will demonstrate a higher preference for actions a_1 and a_3 when responding to the probe stimuli. Conversely, in the conflict condition where color was neglected, the probe stimulus should be perceived as more in line with stimulus y' (**Fig. 4C**, conflict, encoder), which should be also reflected in the response (**Fig. 4C**, conflict, policy). Therefore, human participants would be likely to use a_2 and a_4. In the control condition, given the lack of a dominant rewarding feature, the response to the probe

stimulus should not show a strong preference, being distributed between those for stimuli x and y' (**Fig. 4C**, control, policy).

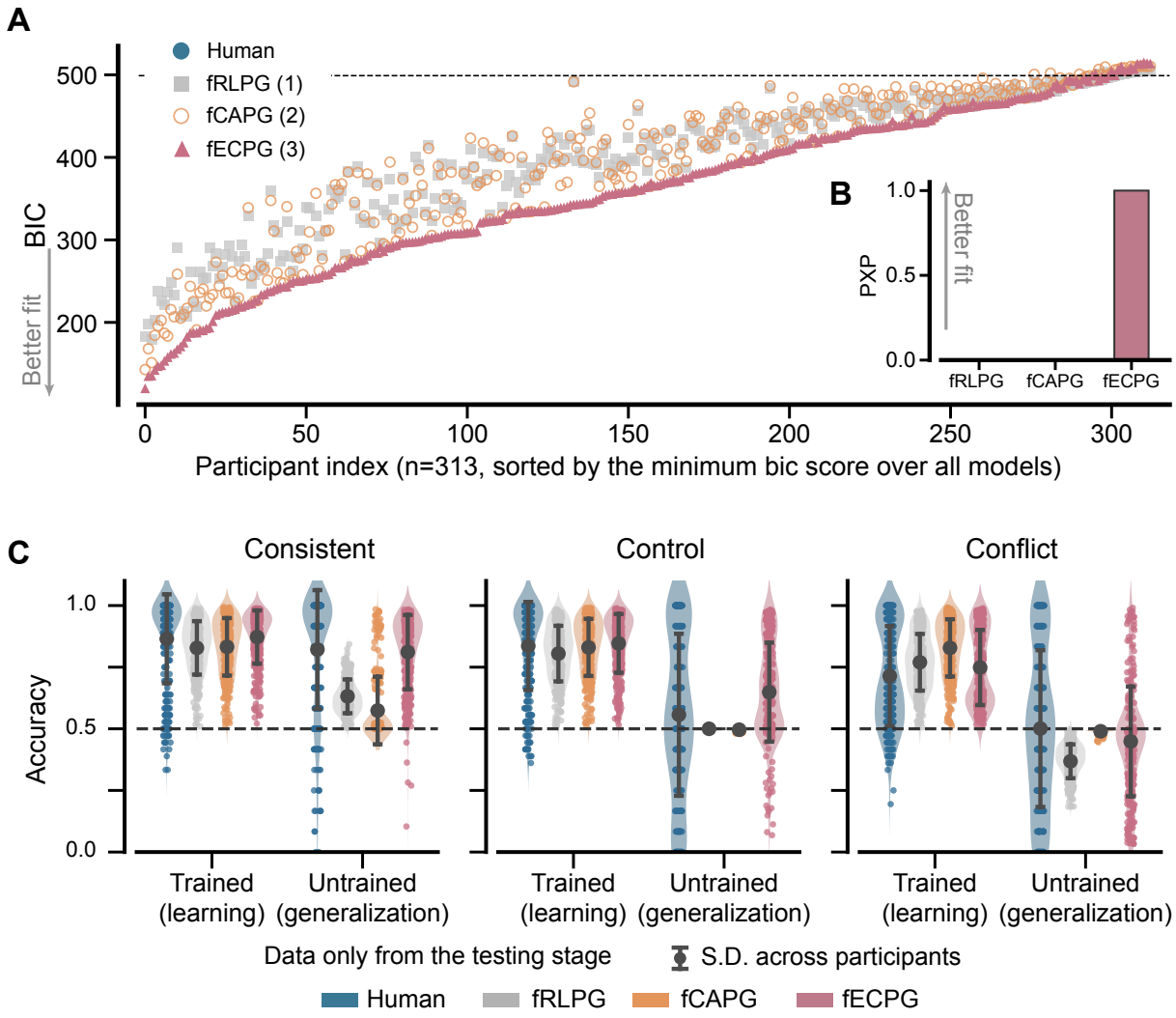

Figure 5 Behaviors of human participants and models in Experiment 2 (part 1). Models were fit to all behavioral data of each participant in both training and testing stages.

A. Models' Bayesian information criterion (BIC) for each participant. The fRLPG model has 1 parameter, the fCAPG model has 2 parameters, and the fECPG model has 3 parameters. The BICs for a fully random policy are represented by dashed lines. Participants with lower BIC scores generally exhibit better learning and generalization performance. See Table 2 for the exact value.

B. Protected exceedance probability (PXP) tallies for each model.

C. Learning and generalization performance of human participants and models for each experimental condition. Dashed lines indicate the 50% chance level. Only data from the testing stage is included in the analysis. Error bars reflect the standard deviation across 313 valid participants.

For the third question, we observed that the efficiency of rewarding feature extraction extends or shortens the time it takes to form stable abstract states, influencing the agent's learning and generalization performance. In the consistent condition, the fECPG model quickly identifies the rewarding feature, enabling it to form stable abstract states rapidly (**Fig. 4D**, consistent), which results in a high degree of generalization (**Fig. 4E**, consistent). However, in the control condition, where no salient cue dominates, the model experiences a slower state abstraction process (**Fig. 4D**, control). In the conflict condition, the need to suppress color prolongs the time required to extract rewarding features, thereby further extending the state abstraction period (**Fig. 4D**, conflict). Consequently, the prolonged state abstraction period reduces the time available for policy decoding, resulting in poorer learning and generalization performance in the conflict case (**Fig. 4E**, control, conflict).

To validate these model predictions, we collected behavioral data from 313 participants who each completed three task blocks corresponding to consistent, control, and conflict conditions. We fit all three feature-based models and found that both BIC and PXP preferred the fECPG model as the best model for capturing human behavioral data, consistent with the findings from Experiment 1 (**Fig. 5A, B** and **Table 2**).

More importantly, as the principle of efficient coding predicts, human participants exhibit different levels of generalization across experimental conditions. They achieved high untrained accuracy in the consistent condition, lower in the control condition, and lowest in the conflict condition (**Fig. 5C**, blue). Beyond the overall trend, humans' generalization behaviors are also characterized by high variability. Some participants generalized effectively across all conditions, while others always negatively transferred their knowledge. This variability is also accurately captured by the fECPG model (**Fig. 5C**, red), but not by the two classical RL models without efficient coding (**Fig. 5C**, gray and orange).

To our surprise, the ECPG model shows a significant advantage in predicting human learning performance—an area where classical RL models have traditionally been preferred. Participants demonstrated a more rapid improvement in the consistent condition than in the control and conflict conditions. Specifically, the learning curve in the conflict condition was

markedly slower when compared to the other conditions. Only the fECPG model captured the significantly slower trend (**Fig. 6A**).

The probe design further validates the human participant's ability to extract rewarding features in a way predicted by the efficient coding principle. Human participants' responses to the probe stimuli were consistent with the fECPG model predictions (**Fig. 6B, C**; Spearman's r > 0.60, p <0.001 for all conditions; see **Method 4.9** for the correlation calculation). In contrast, models without efficient coding, fRLPG and fCAPG, failed to replicate such behavioral patterns (see **Extended Data 5.6** for their probe responses), exhibiting significantly weaker correlations with human behavioral data (**Fig. 6C**).

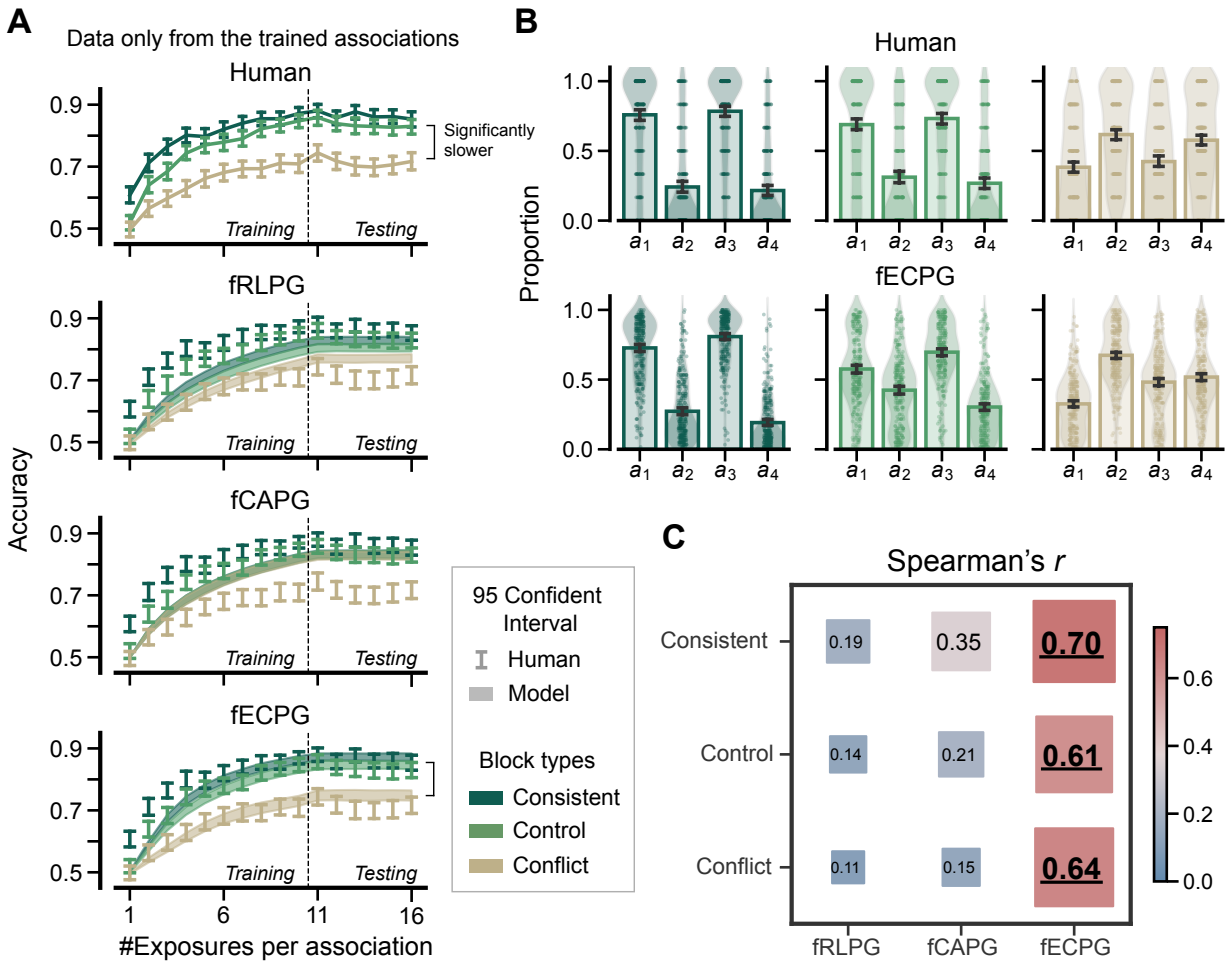

Figure 6 Behaviors of human participants and models in Experiment 2 (part 2). Models were fit to all behavioral data of each participant in both training and testing stages.

A. Proportion of correct responses over the number of times that each association is shown. Dashed lines split the experiment into training and testing stages. The analysis excludes

responses to the untrained associations and the probe stimulus. Error bars indicate the 95% confidence interval for the mean estimate.

B. Humans and the fECPG model respond to the probe stimuli. Error bars indicate the 95% confidence interval for the mean estimate.

C. Correlation between model predictions and human responses to probe stimuli. The annotated value represents Spearman's correlation coefficient under different experimental conditions.

It is important to note that there is a discrepancy between our prediction and human behavior in the control condition. In response to the probe, human participants were likely to use the policy of stimulus x rather than a random policy. This phenomenon could have arisen from two potential factors. First, during training, the experiment might not have adequately balanced the presentation frequency of the stimuli. Participants learned two associations with stimulus x and one with stimulus y' (with the other association tested in the testing stage), which implies that stimulus x was shown twice as frequently as was stimulus y'. Consequently, participants might have adaptively adjusted their encoding and decision-making on these statistics and placed more attention on stimulus x. Second, the color feature might have been inherently more salient to humans. When the three features were equally informative, participants may have naturally prioritized the color feature. However, this gap does not undermine our conclusion that the fECPG model best captures human participants' responses to the probe stimulus.

All evidence leads to one conclusion: during learning, humans strive to distill representations into their simplest and most essential forms. Driven by this goal, human participants learn representations using a small subset of rewarding features within their environments. They further simplify these representations by abstracting them into compact, lower-dimensional internal states, which naturally leads to generalization.

Table 2. Model Fitting in Experiment 2

		fRLPG (1)	fCAPG (2)	fECPG (3)	LC (3)	MA (3)	ACL (5)	fL1PG (3)	fL2PG (3)	fRNDPG (3)	fDCPG (3)
Overall	NLL	196.291	190.74	**170.913**	180.222	176.958	171.557	176.716	172.168	191.569	190.792
	AIC	394.583	385.48	**347.826**	366.445	361.917	353.114	359.433	350.336	389.138	387.585
	BIC	398.469	393.252	**359.484**	378.103	377.461	372.545	371.091	361.994	400.797	399.243
Consistent	NLL	56.465	55.154	47.357	51.416	48.993	**44.073**	48.151	47.412	55.628	55.166
	AIC	114.929	114.307	100.715	108.831	105.986	**98.147**	102.302	100.823	117.255	116.333
	BIC	117.717	119.882	**109.077**	117.194	117.136	112.084	110.664	109.186	125.618	124.695
Control	NLL	64.176	61.41	**56.948**	59.687	57.689	58.525	59.253	57.944	62.399	61.425
	AIC	130.351	126.821	**119.897**	125.375	123.379	127.05	124.506	121.887	130.798	128.851
	BIC	133.139	132.395	**128.259**	133.737	134.529	140.988	132.869	130.25	139.161	137.213
Conflict	NLL	75.651	74.176	**66.607**	69.119	70.276	68.959	69.312	66.813	73.542	74.201
	AIC	153.302	152.352	**139.215**	144.239	148.552	147.917	144.625	139.625	153.085	154.402
	BIC	156.089	157.927	**147.577**	152.601	159.702	161.855	152.987	147.988	161.447	162.764

Bold values indicate the best for each criterion
NLL=negative log likelihood; AIC=Akaike information criterion; Bayesian information criterion

2.4 The human brain optimizes efficient coding to enhance learning and generalization

A potential argument is that the classical RL objective is still a sufficient computational goal to explain human behavior once it provides the necessary cognitive mechanisms at the algorithmic level. We oppose this view for two reasons. First, all current algorithmic-level models with handcrafted cognitive mechanisms fail to capture human behaviors as effectively as the ECPG model. Second, these necessary cognitive mechanisms are inherently designed to simplify representations, essentially pursuing efficient coding.

We developed and compared three algorithmic-level models (**Fig. 7A**). The first model, the *Latent Cause* (LC) model (Gershman et al., 2017; **Method 4.4.7**; see also Collins & Frank, 2013; Collins & Frank, 2016; Gershman et al., 2010) employs a hierarchical nonparametric Bayesian process to simulate human state abstraction ability. During the learning period, the LC model categorizes observed stimuli into latent clusters and learns the decision policy for these clusters. The second model, called the *Memory-Association* (MA) model (**Method 4.4.8**), memorizes all stimuli and their preferred actions, establishing associations between stimuli that share the same actions. These associations facilitate the inference of correct actions in untrained tasks, thereby

enabling generalization. The third model, *Attention at Choice and Learning* (ACL; (Leong et al., 2017); **Method 4.4.9**; see also (Ballard et al., 2018; Niv et al., 2015), learns the value of each feature and calculates the feature importance based on these values. The model uses a linearly weighted feature value for decision-making. Notably, the LC and MA models emphasize state abstraction ability, whereas the ACL model is designed to extract and prioritize rewarding features. Both abilities could emerge by optimizing for the efficient coding goal, but in a different computational formulation.

We tested all three algorithmic-level models on Experiment 2, with a focus on two qualitative metrics: generalization in the control case to examine their latent cause abstraction ability (**Fig. 7C**) and response to probe stimuli to evaluate their rewarding feature extraction (**Fig. 7D**). All three models underperformed the fECPG model in terms of BIC and PXP (**Fig. 7B, Table2**). The LC and MA models fail to account for human responses to probe stimuli (**Fig. 7D**) due to the lack of a feature extraction mechanism. The ACL model struggled with generalizing in the control case (**Fig. 7C**) as well as extracting rewarding features in both the control and conflict cases (**Fig. 7D**), because the feature importance calculated based on the accumulated value cannot deemphasize the negatively-rewarded feature effectively (see **Supplemental Note 2.2** for further discussion). These results underscore the superior performance of the fECPG model, a computational-level model, in modeling human behaviors and support our hypothesis that human participants learn simplified representations when maximizing rewards.

From a machine learning perspective, the fECPG model proposed here addresses a regularized optimization problem. This raises a final question: can the efficient-coding term be substituted by other commonly-used machine learning regularizers? We implemented an *L1-Norm Policy Gradient* (L1PG; **Method 4.4.10**) and an *L2-Norm Policy Gradient* (L2PG; **Method 4.4.10**), incorporating L1 or L2 norms as heuristic approximations for representation complexity. While the L1PG model exhibited a significant performance gap, the L2PG model performed more comparably to the fECPG model, as shown in **Fig. 7B, C, and D**. Although a substantial portion (~36%) of participants were better described by the L2PG model, these participants displayed distinct behavioral dynamics: compared to participants better captured by the fECPG model, they tended to learn more slowly and showed weaker generalization (**see Supplementary Note 2.3**

for details). This suggests that the ECPG model has a unique capability in capturing humans' fast learning and strong generalization patterns. For completeness, we also tested a *Random Regularizer Policy Gradient* (RNDPG; **Method 4.4.11**), which injects noise into the encoder weights, as well as a Decoder Complexity Policy Gradient (DCPG; **Method 4.4.10**), which constrains decoder complexity. However, these models proved ineffective at both generalizing and extracting rewarding features (**Fig. 7B, C, D**).

Finally, we validated our conclusions by performing a model recovery analysis to test our ability to differentiate between models (**Extended Data 5.8**). Importantly, we found that the ECPG model can be uniquely distinguished from the other models. The low false positive rate (with other models unlikely to be misidentified as ECPG) indicates that the ECPG model's superior performance over the control models is not due to its expressiveness but to its accurate description of human behavior. Thus, these findings confirm that our model recovery approach robustly supports our conclusion that the ECPG model, with its augmented RL objective incorporating efficient coding, best accounts for human learning and decision-making.

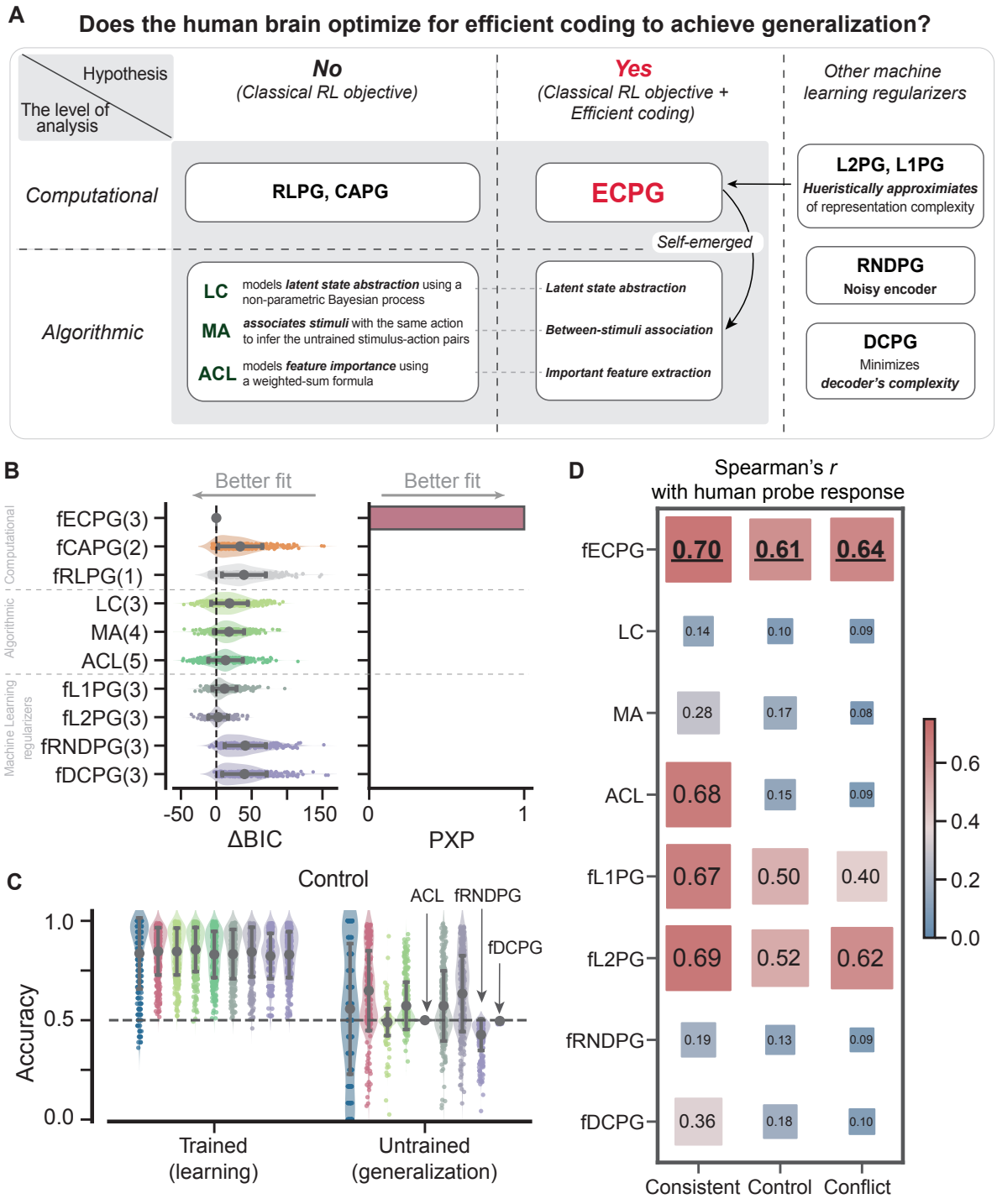

Figure 7 Comparison of the (f)ECPG model with other algorithmic-level models in Experiment 2. Error bars reflect the standard deviation (S.D.).

A. An overview of models across hierarchical levels. The central research question explores whether the human brain optimizes for efficient coding to enhance generalization. At the computational level, the ECPG model affirms this hypothesis ("Yes"), whereas the RLPG and CAPG models represent the opposing viewpoint ("No"). These models collectively represent the entire hypothesis space at the computational level. Below this, the ECPG

model is contrasted with various algorithmic-level models (LC, MA, ACL), each designed with specific cognitive mechanisms. Additionally, the ECPG model is compared against several common machine learning regularizers (L2PG, L1PG, DCPG) that also aim to reduce model complexity, but through different methods.

B. Model comparisons for all models in terms of BIC and PXP in Experiment 2. See **Extended Data 5.5** for model comparison in Experiment 1.

C. Learning and generalization performances across all models in the control case of Experiment 2. See **Extended Data 5.6** for generalizations in other conditions.

D. Correlation between model predictions and human responses to probe stimuli at different experimental conditions. See **Extended Data 5.7** for the bar plots.

3. Discussion

The classical RL framework has limitations in terms of its ability to explain human representation learning and generalization. In this paper, we proposed augmenting the classical RL objective with the efficient coding principle: an intelligent agent should distill the simplest necessary representations that enable it to achieve its behavioral objectives. A computational-level model derived from the revised framework (*efficient coding policy gradient;* ECPG), predicts that an intelligent agent automatically learns to construct representations with a small set of rewarding features with the environment. These representations are further simplified by abstracting them into compact, lower-dimensional internal states, which naturally results in generalization. These predictions were validated in two behavioral experiments, where the ECPG model consistently provided a more accurate description of human behavior than two classical RL models without efficient coding as well as several published human representation learning models. These findings indicate that efficient coding offers a more suitable computational objective in understanding human behavior.

In this paper, we examine whether the classical RL objective alone, or in combination with efficient coding, better aligns with Marr's computational level in explaining human behavior. A potential critique of our approach in **Section 2.4** is the lack of comparison with an alternative model capable of generalizing without representation simplification. However, we found no such model in the existing literature. This absence reflects the historical context of the acquired equivalence paradigm on which our study builds. Although generalization within this paradigm has long been documented since Hall (1991), previous explanations—including categorization (Urcuioli & Vasconcelos, 2008), stimulus association (Shohamy & Wagner, 2008), and selective attention (Bonardi et al., 2005)—are all encompassed by our efficient coding framework. In other words, algorithmic models based on selective attention, for example, inherently implement mechanisms predicted by the computational-level goal of efficient coding.

The proposed ECPG model can potentially serve as a valuable quantitative analytical tool for understanding cognitive impairments in individuals with mental disorders. Previous research using the acquired equivalence paradigm has demonstrated that people with schizophrenia (Farkas et al., 2008; Keri et al., 2005), mild Alzheimer's disease (Bodi et al., 2009), hippocampal

atrophy (Myers et al., 2003), and Parkinson's disease (Myers et al., 2003) exhibit dysfunction in the acquired equivalence task. We believe that the ECPG model, given its detailed portrayal of human learning and generalization in this paradigm, can help elucidate the underlying causes of these cognitive anomalies.

Beyond serving as a better empirical model for human learning, the proposed computational objective could potentially represent a rational strategy (specifically, a resource-rational strategy; see below) for humans. The classical RL objective was designed to maximize expected reward in narrowly defined settings, where agents focus on learning a single, well-defined task (Sutton & Barto, 2018). However, humans live in more complex and dynamic real-world environments, where decision-making requires agents to generalize effectively from past experiences to earn rewards in unseen scenarios. Moreover, the human brain is innately capacity-constrained (Miller, 1956); it has inherent limitations in processing and storing information, which requires the efficient use of cognitive resources. Therefore, learning simpler representations that facilitate generalization is a crucial component in the pursuit of maximizing reward in real-world decision-making. We believe that this insight can also improve learning and generalization in artificial intelligence operating under real-world conditions.

The idea of linking RL to efficient coding has been applied to understand learning and generalization in various contexts (Berger & Machens, 2020; Botvinick et al., 2015; Franklin & Frank, 2020; Frydman & Jin, 2022; Jaskir & Frank, 2023; Luettgau et al., 2023; Xia & Collins, 2021). For example, this approach has been shown to better explain monkeys' neural activity in frontal areas (Berger & Machens, 2020), humans' risky choice behavior (Frydman & Jin, 2022), and meta-level generalization between tasks (Franklin & Frank, 2020). Here, we present a specific formalization of efficient coding using information-theoretic measures. We demonstrate that this approach provides a better empirical description of both human learning and generalization behaviors compared to several alternatives.

Our study bridges the gap between representation learning in the human brain and machine learning. In cognitive science, researchers have applied latent cause clustering (LC) and Association-Choice Learning (ACL) models to understand a variety of phenomena (Radulescu et al., 2021). The latent cause clustering explains Pavlovian conditioning and extinction (Gershman

et al., 2010), memory modification (Gershman et al., 2017), social classification (Gershman & Cikara, 2023), functional-based generalization (Collins & Frank, 2013; Collins & Frank, 2016; Lehnert et al., 2020), etc. The selective attention, on the other hand, is used to explain concept formation (Mack et al., 2016), the evolution of beliefs (Markovic et al., 2015), and has received neural evidence from eye-tracking and functional Magnetic Resonance Imaging (fMRI) studies (Leong et al., 2017; Niv et al., 2015). In machine learning, researchers have focused on how information-theoretic regularizers facilitate an artificial agent performing complex cognitive tasks. For example, information-theoretic regularizers may help an agent learn robust state abstractions that enhance learning speed (Chelombiev et al.; Islam et al., 2022; Konidaris, 2019) and form a simple but informative world model (Ferns & Precup, 2014; Rakelly et al., 2021). Our study demonstrates that in a simple cognitive task, both mechanisms serve the unified objective of minimizing representation complexity, guided by an information-theoretic regularizer. This finding facilitates communication between the two fields and contributes to a unified research framework for understanding both machine and human intelligence. Building on this line of thought, we plan to extend the current framework in future research to more complex task settings, such as multi-step Markov Decision Processes (MDPs), and explore whether complex human behaviors like planning and multi-task learning align with the predictions of information-theoretic regularizers within machine learning.

Recent research has suggested that human intelligence is more accurately described by the principle of *resource-rationality* (Gershman et al., 2015; Griffiths et al., 2015) than by the classical notion of *rationality* (Von Neumann & Morgenstern, 1947). The resource-rationality principle emphasizes the need to consider computational costs in the pursuit of maximum reward, building on the classical notion of rationality. The combination of efficient coding and reward maximization principles applied in this study encapsulates the idea of resource-rationality, with reward maximization representing the notion of rationality and representation complexity representing computational costs. The basic idea is that information transmission in the brain incurs significant metabolic costs, thus minimizing representation complexity (a quantification of average information transmitted into the brain) serves as a reasonable proxy to minimize computational costs (Zenon et al., 2019). Notably, while numerous studies have employed

resource-rationality to explain deviations from pure rationality in human behavior (Gershman, 2020; Lieder & Griffiths, 2020; Sims, 2016), our research further emphasizes the advantages conferred by the principle, particularly in accounting for state abstraction, rewarding feature extraction, and generalization.

Reference

Asadi, A., Abbe, E., & Verdú, S. (2018). Chaining mutual information and tightening generalization bounds. *Advances in Neural Information Processing Systems*, *31*.

Ballard, I., Miller, E. M., Piantadosi, S. T., Goodman, N. D., & McClure, S. M. (2018). Beyond Reward Prediction Errors: Human Striatum Updates Rule Values During Learning. *Cereb Cortex*, *28*(11), 3965-3975.

Barlow, H. B. (1961). Possible principles underlying the transformation of sensory messages. *Sensory communication*, *1*(01), 217-233.

Barto, A. G. (1995). Adaptive critics and the basal ganglia.

Bates, C. J., Lerch, R. A., Sims, C. R., & Jacobs, R. A. (2019). Adaptive allocation of human visual working memory capacity during statistical and categorical learning. *J Vis*, *19*(2), 11.

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: a review and new perspectives. *IEEE Trans Pattern Anal Mach Intell*, *35*(8), 1798-1828.

Berger, S., & Machens, C. K. (2020). Compact task representations as a normative model for higher-order brain activity. *Advances in Neural Information Processing Systems*, *33*, 3209-3219.

Bodi, N., Csibri, E., Myers, C. E., Gluck, M. A., & Keri, S. (2009). Associative learning, acquired equivalence, and flexible generalization of knowledge in mild Alzheimer disease. *Cogn Behav Neurol*, *22*(2), 89-94.

Bonardi, C., Graham, S., Hall, G., & Mitchell, C. (2005). Acquired distinctiveness and equivalence in human discrimination learning: evidence for an attentional process. *Psychon Bull Rev*, *12*(1), 88-92.

Botvinick, M., Weinstein, A., Solway, A., & Barto, A. (2015). Reinforcement learning, efficient coding, and the statistics of natural tasks. *Current opinion in behavioral sciences*, *5*, 71-77.

Browning, M., Behrens, T. E., Jocham, G., O'Reilly, J. X., & Bishop, S. J. (2015). Anxious individuals have difficulty learning the causal statistics of aversive environments. *Nat Neurosci*, *18*(4), 590-596.

Chelombiev, I., Houghton, C., & O'Donnell, C. Adaptive estimators show information compression in deep neural networks. arXiv 2019. *arXiv preprint arXiv:1902.09037*.

Collins, A. G., & Frank, M. J. (2013). Cognitive control over learning: creating, clustering, and generalizing task-set structure. *Psychol Rev*, *120*(1), 190-229.

Collins, A. G. E., & Frank, M. J. (2016). Neural signature of hierarchically structured expectations predicts clustering and transfer of rule sets in reinforcement learning. *Cognition*, *152*, 160-169.

Crowston, K. (2012). Amazon mechanical turk: A research tool for organizations and information systems scholars. Shaping the Future of ICT Research. Methods and Approaches: IFIP WG 8.2, Working Conference, Tampa, FL, USA, December 13-14, 2012. Proceedings,

Daw, N. D., Gershman, S. J., Seymour, B., Dayan, P., & Dolan, R. J. (2011). Model-based influences on humans' choices and striatal prediction errors. *Neuron*, *69*(6), 1204-1215.

Farkas, M., Polgar, P., Kelemen, O., Rethelyi, J., Bitter, I., Myers, C. E., Gluck, M. A., & Keri, S. (2008). Associative learning in deficit and nondeficit schizophrenia. *Neuroreport*, *19*(1), 55-58.

Ferns, N., & Precup, D. (2014). Bisimulation Metrics are Optimal Value Functions. UAI,

Fisher, A., Rudin, C., & 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. *J. Mach. Learn. Res.*, *20*(177), 1-81.

Flesch, T., Juechems, K., Dumbalska, T., Saxe, A., & Summerfield, C. (2022). Orthogonal representations for robust context-dependent task performance in brains and neural networks. *Neuron*, *110*(24), 4212-4219.

Franklin, N. T., & Frank, M. J. (2020). Generalizing to generalize: Humans flexibly switch between compositional and conjunctive structures during reinforcement learning. *PLoS Comput Biol*, *16*(4), e1007720.

Frydman, C., & Jin, L. J. (2022). Efficient coding and risky choice. *The Quarterly Journal of Economics*, *137*(1), 161-213.

Gershman, S. J. (2020). Origin of perseveration in the trade-off between reward and complexity. *Cognition*, *204*, 104394.

Gershman, S. J., Blei, D. M., & Niv, Y. (2010). Context, learning, and extinction. *Psychol Rev*, *117*(1), 197-209.

Gershman, S. J., & Cikara, M. (2023). Structure learning principles of stereotype change. *Psychon Bull Rev*, *30*(4), 1273-1293.

Gershman, S. J., & Daw, N. D. (2017). Reinforcement learning and episodic memory in humans and animals: an integrative framework. *Annual review of psychology*, *68*, 101-128.

Gershman, S. J., Horvitz, E. J., & Tenenbaum, J. B. (2015). Computational rationality: A converging paradigm for intelligence in brains, minds, and machines. *Science*, *349*(6245), 273-278.

Gershman, S. J., Monfils, M. H., Norman, K. A., & Niv, Y. (2017). The computational nature of memory modification. *Elife*, *6*.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Greydanus, S., Koul, A., Dodge, J., & Fern, A. (2018). Visualizing and understanding atari agents. International conference on machine learning,

Griffiths, T. L., Lieder, F., & Goodman, N. D. (2015). Rational use of cognitive resources: levels of analysis between the computational and the algorithmic. *Top Cogn Sci*, *7*(2), 217-229.

Guo, S. S., Zhang, R., Liu, B., Zhu, Y., Ballard, D., Hayhoe, M., & Stone, P. (2021). Machine versus human attention in deep reinforcement learning tasks. *Advances in Neural Information Processing Systems*, *34*, 25370-25385.

Hall, G. (1991). *Perceptual and Associative Learning*. Oxford University Press.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., & Lerchner, A. (2016). beta-vae: Learning basic visual concepts with a constrained variational framework. International conference on learning representations,

Ho, M. K., Abel, D., Correa, C. G., Littman, M. L., Cohen, J. D., & Griffiths, T. L. (2022). People construct simplified mental representations to plan. *Nature*, *606*(7912), 129-136.

Huang, S., & Ontañón, S. (2020). A closer look at invalid action masking in policy gradient algorithms. *arXiv preprint arXiv:2006.14171*.

Islam, R., Zang, H., Tomar, M., Didolkar, A., Islam, M. M., Arnob, S. Y., Iqbal, T., Li, X., Goyal, A., & Heess, N. (2022). Representation learning in deep rl via discrete information bottleneck. *arXiv preprint arXiv:2212.13835*.

Jaskir, A., & Frank, M. J. (2023). On the normative advantages of dopamine and striatal opponency for learning and choice. *Elife*, *12*.

Jiang, Y., Mi, Q., & Zhu, L. (2023). Neurocomputational mechanism of real-time distributed learning on social networks. *Nat Neurosci*, *26*(3), 506-516.

Keri, S., Nagy, O., Kelemen, O., Myers, C. E., & Gluck, M. A. (2005). Dissociation between medial temporal lobe and basal ganglia memory systems in schizophrenia. *Schizophr Res*, *77*(2-3), 321-328.

Konidaris, G. (2019). On the necessity of abstraction. *Current opinion in behavioral sciences*, *29*, 1-7.

Lehnert, L., Littman, M. L., & Frank, M. J. (2020). Reward-predictive representations generalize across tasks in reinforcement learning. *PLoS Comput Biol*, *16*(10), e1008317.

Leong, Y. C., Radulescu, A., Daniel, R., DeWoskin, V., & Niv, Y. (2017). Dynamic Interaction between Reinforcement Learning and Attention in Multidimensional Environments. *Neuron*, *93*(2), 451-463.

Lerch, R., & Sims, C. R. (2021). Modeling associative motor learning through capacity-limited reinforcement learning. *Journal of Vision*, *21*(9), 2782-2782.

Li, F., Fergus, & Perona. (2003). A Bayesian approach to unsupervised one-shot learning of object categories. proceedings ninth IEEE international conference on computer vision,

Li, L., Walsh, T. J., & Littman, M. L. (2006). Towards a unified theory of state abstraction for MDPs. AI&M,

Lieder, F., & Griffiths, T. L. (2020). Resource-rational analysis: Understanding human cognition as the optimal use of limited computational resources. *Behavioral and brain sciences*, *43*, e1.

Lu, X., Lee, K., Abbeel, P., & Tiomkin, S. (2020). Dynamics generalization via information bottleneck in deep reinforcement learning. *arXiv preprint arXiv:2008.00614*.

Luettgau, L., Erdmann, T., Veselic, S., Stachenfeld, K., Moran, R., Kurth-Nelson, Z., & Dolan, R. (2023). Decomposing dynamical subprocesses for compositional generalization.

Mack, M. L., Love, B. C., & Preston, A. R. (2016). Dynamic updating of hippocampal object representations reflects new conceptual knowledge. *Proceedings of the National Academy of Sciences*, *113*(46), 13203-13208.

Markovic, D., Glascher, J., Bossaerts, P., O'Doherty, J., & Kiebel, S. J. (2015). Modeling the Evolution of Beliefs Using an Attentional Focus Mechanism. *PLoS Comput Biol*, *11*(10), e1004558.

Meeter, M., Shohamy, D., & Myers, C. E. (2009). Acquired equivalence changes stimulus representations. *J Exp Anal Behav*, *91*(1), 127-141.

Miller, G. A. (1956). The magical number seven plus or minus two: some limits on our capacity for processing information. *Psychol Rev*, *63*(2), 81-97.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., & Ostrovski, G. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529-533.

Montague, P. R., Dayan, P., & Sejnowski, T. J. (1996). A framework for mesencephalic dopamine systems based on predictive Hebbian learning. *J Neurosci*, *16*(5), 1936-1947.

Myers, C. E., Shohamy, D., Gluck, M. A., Grossman, S., Onlaor, S., & Kapur, N. (2003). Dissociating medial temporal and basal ganglia memory systems with a latent learning task. *Neuropsychologia*, *41*(14), 1919-1928.

Nelli, S., Braun, L., Dumbalska, T., Saxe, A., & Summerfield, C. (2023). Neural knowledge assembly in humans and neural networks. *Neuron*, *111*(9), 1504-1516 e1509.

Niv, Y. (2009). Reinforcement learning in the brain. *Journal of Mathematical Psychology*, *53*(3), 139-154.
Niv, Y. (2019). Learning task-state representations. *Nat Neurosci*, *22*(10), 1544-1553.
Niv, Y., Daniel, R., Geana, A., Gershman, S. J., Leong, Y. C., Radulescu, A., & Wilson, R. C. (2015). Reinforcement learning in multidimensional environments relies on attention mechanisms. *J Neurosci*, *35*(21), 8145-8157.
Niv, Y., & Langdon, A. (2016). Reinforcement learning with Marr. *Curr Opin Behav Sci*, *11*, 67-73.
Noh, H., You, T., Mun, J., & Han, B. (2017). Regularizing deep neural networks by noise: Its interpretation and optimization. *Advances in Neural Information Processing Systems*, *30*.
Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., & Antiga, L. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, *32*.
Pensia, A., Jog, V., & Loh, P.-L. (2018). Generalization error bounds for noisy, iterative algorithms. 2018 IEEE International Symposium on Information Theory (ISIT),
Rac-Lubashevsky, R., Cremer, A., Collins, A. G. E., Frank, M. J., & Schwabe, L. (2023). Neural Index of Reinforcement Learning Predicts Improved Stimulus-Response Retention under High Working Memory Load. *J Neurosci*, *43*(17), 3131-3143.
Radulescu, A., Shin, Y. S., & Niv, Y. (2021). Human Representation Learning. *Annu Rev Neurosci*, *44*, 253-273.
Rakelly, K., Gupta, A., Florensa, C., & Levine, S. (2021). Which Mutual-Information Representation Learning Objectives are Sufficient for Control? *Advances in Neural Information Processing Systems*, *34*, 26345-26357.
Rescorla, R. A. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and non-reinforcement. *Classical conditioning, Current research and theory*, *2*, 64-69.
Ribas-Fernandes, J. J., Solway, A., Diuk, C., McGuire, J. T., Barto, A. G., Niv, Y., & Botvinick, M. M. (2011). A neural signature of hierarchical reinforcement learning. *Neuron*, *71*(2), 370-379.
Rigoux, L., Stephan, K. E., Friston, K. J., & Daunizeau, J. (2014). Bayesian model selection for group studies—revisited. *Neuroimage*, *84*, 971-985.
Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, *20*, 53-65.
Schultz, W., Dayan, P., & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science*, *275*(5306), 1593-1599.
Shepard, R. N. (1987). Toward a universal law of generalization for psychological science. *Science*, *237*(4820), 1317-1323.
Shohamy, D., & Wagner, A. D. (2008). Integrating memories in the human brain: hippocampal-midbrain encoding of overlapping events. *Neuron*, *60*(2), 378-389.
Shwartz-Ziv, R. (2022). Information flow in deep neural networks. *arXiv preprint arXiv:2202.06749*.
Silver, D., Singh, S., Precup, D., & Sutton, R. S. (2021). Reward is enough. *Artificial Intelligence*, *299*, 103535.
Simoncelli, E. P., & Olshausen, B. A. (2001). Natural image statistics and neural representation. *Annu Rev Neurosci*, *24*, 1193-1216.
Sims, C. R. (2016). Rate-distortion theory and human perception. *Cognition*, *152*, 181-198.

Sims, C. R. (2018). Efficient coding explains the universal law of generalization in human perception. *Science*, *360*(6389), 652-656.
Sims, C. R., Jacobs, R. A., & Knill, D. C. (2012). An ideal observer analysis of visual working memory. *Psychol Rev*, *119*(4), 807-830.
Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.
Tishby, N., Pereira, F. C., & Bialek, W. (2000). The information bottleneck method. *arXiv preprint physics/0004057*.
Tomov, M. S., Schulz, E., & Gershman, S. J. (2021). Multi-task reinforcement learning in humans. *Nat Hum Behav*, *5*(6), 764-773.
Urcuioli, P. J., & Vasconcelos, M. (2008). Effects of within-class differences in sample responding on acquired sample equivalence. *J Exp Anal Behav*, *89*(3), 341-358.
van Opheusden, B., Kuperwajs, I., Galbiati, G., Bnaya, Z., Li, Y., & Ma, W. J. (2023). Expertise increases planning depth in human gameplay. *Nature*, *618*(7967), 1000-1005.
Von Neumann, J., & Morgenstern, O. (1947). Theory of games and economic behavior, 2nd rev.
Wei, X. X., & Stocker, A. A. (2015). A Bayesian observer model constrained by efficient coding can explain 'anti-Bayesian' percepts. *Nat Neurosci*, *18*(10), 1509-1517.
Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, *8*, 229-256.
Xia, L., & Collins, A. G. E. (2021). Temporal and state abstractions for efficient learning, transfer, and composition in humans. *Psychol Rev*, *128*(4), 643-666.
Zenon, A., Solopchuk, O., & Pezzulo, G. (2019). An information-theoretic perspective on the costs of cognition. *Neuropsychologia*, *123*, 5-18.
Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., & Oliva, A. (2014). Learning deep 1143 features for scene recognition using places database. *Z. Ghahramani*, *1144*.

4. Method

4.1 Ethics statement

Participants were provided informed consent. The experimental protocol was approved by the University Committee on Activities involving Human Subjects at Rensselaer Polytechnic Institute (IRB-2055). Our experiment did not collect any demographical information from participants, including gender.

4.2 Materials

The experiments were designed based on the paradigm of acquired equivalence (AE) (Meeter et al., 2009; Myers et al., 2003; Shohamy & Wagner, 2008). The experiment included two types of pictures: *alien* and *scene*. For the alien pictures, we utilized the "greebles" stimuli created by Michael Tarr[1]. The original greeble stimuli are purple. We created several new greeble stimuli by changing their color. Regarding the scene pictures, we sampled from the "Places205" picture database, as reported in Zhou et al., (2014)[2].

In the main experiment blocks, aliens and scenes were organized into sets. Each set comprised four alien stimuli (x, x', y, y') and four scenes (a_1, a_2, a_3, a_4), resulting in eight unique associations per set. Six of these associations were trained during the association stage, while all eight were tested in the testing stage. It is important to note that the stimuli in the AE task are defined to be "superficially dissimilar". In our experiment, the greeble stimuli within a block were required to have the same color but exhibit mutually different shapes and appendages. There was no correlation between the alien's shape (the configuration of its appendages) and the correct response.

4.3 Experiments

4.3.1 Experiment 1

We recruited 302 participants from Amazon Mechanical Turk (MTurk) (Crowston, 2012). All participants provided informed consent before starting the experiment. Each participant

[1] Greebles stimuli of Michael J Tarr, Carnegie Mellon University. Downloaded from http://www.tarrlab.org/

[2] Downloaded from http://places.csail.mit.edu/downloadData.html.

completed two practice blocks. To ensure a comprehensive understanding of the experiment, participants were required to achieve at least 70% accuracy in the second practice block to progress to the main experimental stage. Those who did not meet this criterion were allowed to repeat the second practice block until they achieved the necessary performance level; otherwise, they could not proceed to the main experiment. Participants received a base payment of $2 plus a bonus of up to $3 based on their response accuracy.

This project aimed to study generalization within the learning process, meaning participants who did not learn were outside the scope of this study. Consequently, we excluded 137 participants who failed a screening criterion (average accuracy lower than 60% for the last 24 trials, equating to 4 repetitions, in the training stage). All analyses in Experiment 1 were conducted with the remaining 165 qualified participants.

The experiment consisted of two types of trials: training and testing. Each training trial comprised three screens. Following a 500 ms fixation screen, the trial presented an alien stimulus in the upper middle of the screen, along with photographs of scenes, offering one correct and one incorrect choice. Participants were instructed, “Which scene is associated with this alien?” and asked to respond by pressing the “F” or “J” key. These choices’ left-right order was counterbalanced across trials. The stimulus screen remained visible for ten seconds, followed by a one-second feedback screen displaying either “Correct! You got 1 point.” or “Incorrect! You got 0 point.” The test stage trials were identical, except that no feedback was provided after responses. In the testing stage, the experiment was directed to the next fixation screen following the participant's response or after a maximum duration of ten seconds.

Experiment 1 included two experimental blocks. Each block consisted of a training stage, during which participants learned the stimulus-action associations, and a testing stage, during which participants were tested on the learned associations as well as an untrained generalization probe (the dashed associations). The training stage involved each association being trained ten times with feedback, resulting in 6 (associations) × 10 (repetitions) = 60 training trials. The testing stage tested both the trained and untrained associations six times, resulting in 8 (associations) × 6 (repetitions) = 48 testing trials. Participants were explicitly informed of the transition between

the two experimental stages, and they were also reminded to keep and reapply their training experiences to achieve better performance.

Before the main blocks, each participant was required to complete two practice blocks. The first practice block contained a simple trial-and-error learning task, where participants were trained to learn the correct answer through feedback. They were asked to correctly associate $x - a_1$ and $y - a_2$ without being asked to build any between-stimuli equivalence. This block provided a gentle introduction to the experiment, with ten trials and unlimited response time. The primary goal of the first practice was to familiarize participants with the trial-and-error training process. The second practice served as a quiz. This block included a simplified version of the main training stage, where participants were presented with four stimuli but only required to choose from two actions. It contained 4 (associations) × 10 (repetitions) = 40 training trials. Participants needed to achieve 70% accuracy to pass the quiz; otherwise, they were required to repeat the second practice block before progressing to the main experimental blocks. The practice blocks were designed to help participants learn to establish between-stimuli equivalence in preparation for the main experimental blocks and used similar materials as the main blocks.

4.3.2 Experiment 2

We recruited 497 participants from MTurk. All participants gave informed consent prior to the experiment. Each participant completed two practice blocks. To ensure full understanding of the experiment, they needed to achieve at least 70% accuracy in the second practice block to proceed to the main experimental stage. Those who did not meet this criterion were given the opportunity to repeat the second practice block until they reached the required accuracy. All participants received a $3 base payment plus up to a $4.5 bonus based on their response accuracy.

We filtered the participants' data using the same screening criterion as in Experiment 1. A total of 184 participants were excluded because they did not achieve an average accuracy of 60% for the last 24 trials (equivalent to 4 repetitions) in the training stage. All analyses in Experiment 2 were conducted with the remaining 313 qualified participants.

Note that Experiment 2 included twice as many qualified participants as Experiment 1. This is because each participant in Experiment 1 completed two identical experimental blocks, while in Experiment 2, participants completed three different blocks, each corresponding to a different experimental condition. To ensure that each condition in Experiment 2 had a comparable amount of data to Experiment 1, we increased participant enrollment.

After completing the same practice blocks as in Experiment 1, participants were required to complete three main experimental blocks: a consistent block, a control block, and a conflict block. The sequence of these blocks was counterbalanced among participants. The three blocks were almost identical; the only difference lay in the stimuli's appearance.

Within each block, participants were required to complete a 60-trial training stage, which was the same as in Experiment 1. They then entered the testing stage, where they had to respond to eight regular testing associations plus an additional probe stimulus. Consequently, the testing stage comprised 9 (associations) × 6 (repetitions) = 54 trials.

The remaining details of Experiment 2 were identical to Experiment 1. Note that, unlike the untrained associations, we did not predefine a correct answer for the probe stimulus. We simply record participants' responses and hope to uncover which feature people were attending to by analyzing the response distribution.

4.4 Models

To set the stage, we first formalize a dynamic decision process in the AE paradigm. For consistency, we adopt a notation system similar to that used in the experimental paradigm.

We refer to a participant or decision maker as an *agent*. In each trial t, an agent is presented with an alien stimulus s_t from the set $\{x, x', y, y'\}$. The agent's task is to select an action, specifically a scene picture a_t, from the set $\{a_1, a_2, a_3, a_4\}$, with the objective of maximizing the reward $r(s_t, a_t)$ based on the feedback received. The subscript t denotes the variable at a particular trial. Both the stimulus S and action A are defined as categorical variables.

4.4.1 RLPG : Reinforcement learning policy gradient

RLPG is a computational level model. The goal of the RLPG model is to identify a policy π that optimizes the classical RL goal.

$$\max_{\pi} E_{\pi}\left[r(s_t, a_t)\right] \tag{4}$$

In the AE experiment, an agent was required to choose from two possible actions; before receiving any feedback, each action had a 50% chance of being correct. The agent should have had a baseline estimation of reward, denoted as b, prior to making a decision. An action is considered positive when it yields a reward higher than the baseline and negative when the reward is lower. We revised Eq. 4 to include this baseline reward estimation b ,

$$\max_{\pi} E_{\pi}\left[r(s_t, a_t) - b\right] \tag{5}$$

The formula indicates that the RL baseline learns to adjust the policy $\pi(a|s)$ to maximize received reward subtracted by the baseline $r(s_t, a_t) - b$. The reward subtracted by the baseline is commonly called *advantage* in the machine learning community. In this AE task, we assumed b = 0.5, corresponding to an expected reward of 0.5 (reward of 1 with 50% probability).

The objective function can theoretically be tackled by any RL algorithm, but we have chosen the simplest: the policy gradient method. We assume the policy follows a parameterized softmax distribution, transforming the optimization problem into a parameter search:

$$\pi(a|s;\phi) = \frac{\exp[\phi(s,a)]}{\sum_{a'}\exp[\phi(s,a')]} \tag{6}$$

where ϕ denotes the parameters of the policy. Here, ϕ is a 4-by-4 table (4 stimuli by 4 action). $\phi(s,a)$ refer the parameter a row s and column a (See **Extended Data 5.2** for model architecture). For simplicity, we will denote this softmax formula as $\mathrm{softmax}(\phi(s,a))$.

Let $J(\phi) = \max_{\phi} E[r(s_t, a_t) - b]$, then the policy parameters were updated based on the gradient of the objective function $\nabla_{\phi} J(\phi)$,

$$\phi(s,a) = \ \phi(s,a) + \alpha_{\pi}\nabla_{\phi} J(\phi)(s,a) \tag{7}$$

where $\alpha_{\pi} \geq 0$ is the learning rate of policy π. This policy learning rate is the only parameter in the RL baseline model. The policy parameters ϕ were initialized to 0 before the experiment. Eq. 7 updates the policy via its gradient, which gives the name “policy gradient”. We have derived the analytical gradient for both models and verified the derivation using `pyTorch` package

(Paszke et al., 2019). See supplementary material for detailed derivation. The RLPG model features a single parameter: the policy's learning rate, α_π.

There are two remarks related to this simple model. First, though not explicitly shown, the RLPG assumes a perfect representation that fully reconstructs the stimulus. If we construct a model that explicitly includes the representation z and assume that each stimulus s deterministically maps to a unique representation z, the model nevertheless collapses to the RLPG model described above. Second, the RLPG model introduced in this study behaves similarly to the classic Q-learning model, extensively used in psychology (Daw et al., 2011). The most significant advantage of RLPG is its simplicity. The model has a single learning rate parameter, simultaneously approximating the effects of both the "learning rate" and "inverse temperature" parameters in the classic model. This allows for a more effective distillation of the computational essence underlying representation compression.

4.4.2 ECPG: efficient coding policy gradient

The ECPG model is designed with a dual computational goal: to maximize reward while minimizing representation complexity.

$$\max_{\psi,\rho} E_{\psi,\rho}\left[r(s_t, a_t) - b\right] - \lambda I^{\psi}(S; Z) \tag{8}$$

The parameter $\lambda \geq 0$, referred to as the *simplicity parameter*, controls for the tradeoff between the classical RL objective and representation simplicity. When $\lambda = 0$, the agent does not compress stimuli representations for simplicity, focusing solely on reward maximization. Conversely, as $\lambda \to \infty$, the agent learns the simplest set of representations, encoding all stimuli into a single, identical representation. Therefore, an optimal λ balances compression and oversimplification.

The introduction of latent representation z divides the policy into an encoder, ψ, and a decoder, ρ, both of which are optimized according to Eq. 8. Like the RLPG, we solve Eq. 8 using the policy gradient. Here, we considered a parameterized softmax encoder $\psi(z|s;\theta)$ and a decoder $\rho(a|z;\phi)$. The encoder parameter θ is a 4-by-4 table (4 stimuli by 4 representations) and the decoder parameter ϕ is also a 4-by-4 table (See **Extended Data 5.2** for model architecture). The policy π is derived from the combination of the encoder and decoder:

$$\pi(a_t|s_t) = \sum_z \phi(z|s_t;\theta)\rho(a_t|z;\phi) \tag{9}$$

We iteratively update the encoder and decoder to optimize Eq. 8 using the following scheme:

$$\begin{cases} \max\limits_{\theta} J(\theta) = \max\limits_{\theta} \sum\limits_{z} \psi(z|s_t;\theta) \left[\rho(a_t|z)(r(s_t,a_t) - b) - \lambda \log \dfrac{\psi(z|s_t;\theta)}{p(z)} \right] \\ \max\limits_{\phi} J(\phi) = \max\limits_{\phi} \sum\limits_{z} \rho(a_t|z;\phi)[\psi(z|s_t)(r(s_t,a_t) - b)] \\ p(z) \quad = \sum\limits_{s} \psi(z|s)p(s) \end{cases} \tag{10}$$

Here, $p(z)$ indicates the prior preference to the representation z. The first two optimization problems were solved using gradient ascent with learning rate parameters, α_ψ and α_ρ. The prior representation probability $p(z)$ was updated according to the definition of marginal probability. In practice, we also experimented with updating the prior in the gradient formula but found it made no significant difference in modeling human behavior. Therefore, we adopted the current scheme to reduce the number of free parameters.

The initialization of the representation variable z is critical. In this article, z is a categorical variable that shares the same sample space as the stimulus. The encoder parameters θ are initialized by passing the product of an identity matrix and an initial value through a softmax function,

$$\psi(z|s;\theta_0) = \mathrm{softmax}\big(\theta_0 \mathbf{I}(s,z)\big) \tag{11}$$

where is an identity matrix $\mathbf{I}(s,z) = 1 \text{ if } z = s \text{ otherwise } 0$. We pretrained the encoders to reach 99% discrimination accuracy by tuning the initial value θ_0. The initial encoder in this case encoded stimuli almost orthogonally, reflecting the "superficially dissimilarity" in the definition of AE. See **Method 4.5** for more details.

In summary, the ECPG model has three parameters: an encoder learning rate α_ψ, a decoder learning rate α_ρ, a simplicity parameter λ. The stimulus encoder parameters were initialized through pretraining, and the parameters of the decoder were initialized to 0.

The ECPG model's encoder acts as a generative component, similar to the encoder in the beta variational autoencoder (βVAE) as described by Higgins et al., (2016), but with a categorical hidden layer instead of a continuous Gaussian distribution. This design facilitates the

computation of mutual information and the quantification of representation complexity, building upon the work of Lu et al., (2020).

4.4.3 CAPG: the intermediary model

The cascade policy gradient (CAPG) model is a special case of the ECPG model, with the simplicity fixed at 0, $\lambda = 0$. Therefore, the model has only two parameters: the learning rate for the encoder α_ψ and the learning rate for the decoder α_ρ.

4.4.4 fRLPG: the feature-based RLPG model

The feature-based models we developed are extensions of the models previously introduced. The primary difference lies in the incorporation of a feature embedding function $\mathcal{F}$ that maps a stimulus s onto a set of features f. We crafted a feature embedding function to decompose a greeble stimulus into three distinct features: shape, color, and appendage, using "one-hot encoding" for clear differentiation. For instance, the color "purple" is represented as [1, 0, 0, 0, 0] and "yellow" as [0, 1, 0, 0, 0]. The feature function $\mathcal{F}$ each input stimulus with its one-hot code and concatenates these codes into a 15-dimensional vector f, which serves as the model's input (See **Extended Data 5.2C**).

We modified the policy of the fRLPG model to create a feature-based baseline RL model that does not compress stimuli. This model proposes that visual similarity alone could account for human generalization performance, without the need for a representation compression mechanism. With the feature embedding function $\mathcal{F}$ defined, the policy at trial t can be expressed as follows,

$$\pi(a|\mathcal{F}(s_t);\phi) = \mathrm{softmax}\big(\phi(f_t, a)\big) \tag{12}$$

The parameter ϕ now is a 15-by-4 table. As before, the parameter ϕ was updated using the policy gradient method Eq. 7, and actions are selected by sampling from the softmax policy, Eq. 12. The model has one parameter, the learning rate of policy α_π.

4.4.5 fECPG: the feature-based ECPG model

We modified the ECPG encoder to include the feature embedding function and preserved the previous ECPG decoder formulation (See **Extended Data 5.2D**),

$$\psi(z|\mathcal{F}(s_t);\theta) = \mathrm{softmax}\big(\theta(f_t, z)\big) \tag{13}$$

The parameter θ now is a 15-by-4 table. The decoder $\rho(a|z;\phi)$ remains unchanged.

A new challenge we faced was initializing the encoder parameters for the feature-based model. The previous method, which relied on an identity matrix, was no longer suitable because stimuli with overlapping features naturally appear more similar. For instance, a purple greeble should be more similar to another purple greeble than to a yellow one.

To address this, we introduced a new initialization technique:

First, we measured the visual similarity between a stimulus s and all possible stimuli z (including stimuli s itself) by calculating the dot product of their feature embeddings,

$$\mathrm{sim(s, z)} = \langle \mathcal{F}(s), \mathcal{F}(z) \rangle \qquad \forall z \in \{x, x', y, y'\} \tag{14}$$

Second, we multiplied these similarity scores by a scalar θ_0 and passed them through a softmax function to form the representation of stimuli s,

$$\bar{\psi}(z|s) = \mathrm{softmax}\big(\theta_0 \mathrm{sim}(s, z)\big) \tag{15}$$

As before, the initial value θ_0 is tuned through pretraining. This value controls the perceived similarity between stimuli. When θ_0 is small, stimuli with overlapping features look similar.

Finally, we used the representation $\bar{\psi}(z|s)$ as supervised labels to train the model encoder $\psi(z|\mathcal{F}(s), \theta)$ by minimizing their cross-entropy loss

$$\min_{\theta} -\sum_{z} \bar{\psi}(z|s) \log \psi(z|\mathcal{F}(s), \theta) \tag{16}$$

Once the training has converged, the encoder for the fECPG model is considered initialized. The subsequent learning and decision-making processes are consistent with the original ECPG model.

The model has three parameters $\{\alpha_\psi, \alpha_\rho, \lambda\}$: the learning rate for the encoder α_ψ, the learning rate decoder α_ρ, and the simplicity parameter λ.

4.4.6 fCAPG: the feature-based CAPG model

The fCAPG model is special case of the fECPG with the simplicity parameter fix to 0, $\lambda = 0$. The model has two parameters α_ψ and α_ρ: the learning rate for the encoder α_ψ, the learning rate for decoder α_ρ.

4.4.7 LC: latent-cause clustering

The LC model is an algorithmic-level model, adopted and modified from Gershman et al., (2017). The central idea of the LC model is to use a non-parametric Bayesian process to model the cognitive process of latent-cause clustering. The original model cannot be directly applied to the acquired equivalence task, as it is a model about association learning, not instrumental learning. We modified it to incorporate a value function of latent-cluster and action $Q(z, a)$. To facilitate derivation, we rewrite this value function in the probability format $p(q|z, a)$.

The heart of the LC model lies in learning a Chinese restaurant process to model the human latent-cause clustering process. To construct the LC model, we rely on the following five constructs:

- τ: a weight parameter that decides the influence of a variable at time l to the current variable t, the weight follows a power function,

$$\tau_l = \frac{1}{t-l}, \qquad \forall l \leq t-1 \tag{17}$$

- $p(z|\mathbf{z}_{1:t-1})$: the Chinese restaurant prior of the latent-cause z:

$$p(z=k|\mathbf{z}_{1:t-1}) = \begin{cases} \dfrac{\sum_{l<t-1}\tau_l \mathbf{I}(z_l=k)}{\sum_{l<t-1}\tau_l + \alpha}, \text{if } k \text{ is a old cause} \\ \dfrac{\alpha}{\sum_{l<t-1}\tau_l + \alpha}, \quad \text{if } k \text{ is a new cause} \end{cases} \tag{18}$$

  where $\mathbf{I}(\cdot)$ is an indicator function that returns 1 if the condition is satisfied otherwise 0. α is the concentration parameter of the Chinese restaurant process. In the following derivation, we will also use $p(z)$ to represent this component for short.

- $p(s_t|z)$: the likelihood of the current stimulus given history:

$$p(s_t|z) = \sum_f \mathbf{I}(f = \mathcal{F}(s))p(f|z) \tag{19}$$

The feature f has d dimension ($d = 3$ in this study stands for shape, color, appendage). Each dimension represents an index j of the feature (e.g. $f_2 = 1$ means yellow color). We slightly abused the notation $\mathcal{F}(s_t)$. The probability mass function of a feature vector f is written as,

$$p(f_1, \dots, f_d | z = k) = \frac{\prod_d \prod_j \lambda_{kd}^{\mathbf{I}(f_d=j)} + p}{\sum_d \prod_d \prod_j \lambda_{kd}^{\mathbf{I}(f_d=j)} + p} \tag{20}$$

where d is the dimension of feature vector f. p is the prior for the multi-categorical distribution and is always set to 0.1, $p = 0.1$, in this study. This likelihood distribution is learned throughout training by,

$$\lambda_{kd} = \frac{\sum_{l<t-1} \mathbf{I}(f_d = j)}{\sum_{d',j'} \sum_{l<t-1} \mathbf{I}(f_{d'} = j)} \tag{21}$$

- $p(q|z, a_t)$: the value of latent-cluster and action

$$p(q|z, a_t) = \mathbf{I}(q = \mathbf{w}_{z,a_t}) = Q(z, a_t) \tag{22}$$

The weight $\mathbf{w}$ is always initialized as $1/|A|$.

- $p(r_t|q)$: the reward distribution given q value:

$$p(r_t|q) = q^r (1 - q)^{1-r} \tag{23}$$

This is a Bernoulli distribution, given that the experiment yields a deterministic reward of 0 or 1.

The LC model's learning procedure can be framed as a structural learning problem, which is addressed by the Expected-Maximization (EM) algorithm. See **Supplemental Note 1** for derivation.

- E-step: $p(z|s_t, a_t, r_t; \mathbf{w}) \propto \sum_q p(z)p(s_t|z)p(q|z, a_t)p(r_t|q)$
- M-step: $\mathbf{w} = \mathbf{w} + \eta p(z|s_t, a_t, r_t; \mathbf{w})\big(r - Q(z, a_t)\big)\mathcal{F}(s_t)$

where η is the learning rate.

The LC model needs to make two decisions, assigning the current stimulus s_t to a latent-cluster z_t and picking an action a_t for current stimulus s_t. The latent-cluster z_t is selected to maximize the posterior of the latent-cluster,

$$z_t = \arg\max_z p(z|s_t, a_t, r_t; \mathbf{w}) \tag{24}$$

The stimulus value is passed through a softmax to generate the action. The stimulus value is determined by marginalizing all possible latent causes.

$$\pi(a|s_t) = \text{softmax}\left(\beta \sum_z p(z)p(s_t|z)Q(z,a_t)\right) \tag{25}$$

where β is the inverse temperature.

The four parameters $\{\alpha, \eta, \beta, w_0\}$ of the LC are: the concentration parameter for Chinese restaurant process α, the learning rate η, the inverse temperature of the softmax β, and the initial weight for the value function w_0.

4.4.8 MA: memory-association model

The MA model is an algorithmic-level model that combines memory and association mechanisms. It memorizes stimuli-action pairs and forms associations between stimuli with shared actions or features, using these associations to infer actions for untrained tasks. Stimuli with salient shared features (like “color”) are more easily associated.

During training, the MA model utilizes a feature-action Q value table to store and update the value of stimuli-action pairs based on prediction errors (similar to classical RL models). The update formula is as follows:

$$Q(f,a_t) = Q(f,a_t) + \alpha_Q \delta_t, \qquad \forall f \in \mathcal{F}(s) \tag{26}$$

where the value stimulus action pair is calculated using $Q_s(s,a) = \sum_{f\in\mathcal{F}(s)} Q(f,a_t)$.

The model also maintains an association table $w(s=i, s=j)$, that records the associations between pairs of stimuli based on shared actions. The update rule for this association strength is:

$$w(i,j) = w(i,j) + \alpha_{\text{assoc}}\big(1+k(i,j)\big)\big(\overline{w}(i,j) - w(i,j)\big) \tag{27}$$

where the update target $\overline{w}(i,j) = \sum_{T(s,\,a')=1} Q_s(i,a')Q_s(j,a')$ evaluates the similarity of the action preferences between i and j within the trained association, and $T(s,a)$ is another table that memories whether a certain association $s-a$ has been trained before. k here measures the perceptual similarity between i and j using the dot product of their feature vectors, $k(i,j) = \langle \mathcal{F}(i), \mathcal{F}(j) \rangle$. α_{assoc} is the learning rate for the stimuli association matrix. Note that we weighted

the association update using $1 + k(i, j)$ rather than $k(i, j)$ to ensure that the association update is not zero. The self-association is always 1, $k(i, i) = 1$.

In the decision-making stage, the MA model first needs to decide whether the presented stimulus-action pair (s_t, a) has been trained before. For the trained association pairs, the model uses the learned state-action values for decision-making, while for the untrained pairs, the model infers their values based on the state-action values for all stimuli weighted by their associations $w(s_t, s)$,

$$Q_s(s_t, a) = \begin{cases} \sum_{f \in \mathcal{F}(s_t)} Q(f, a_t), & \text{if } T(s_t, a) = 1 \\ \sum_s \phi(s|s_t, w) \sum_{f \in \mathcal{F}(s_t)} Q(f, a_t), & \text{otherwise} \end{cases} \tag{28}$$

where $\phi(s|s_t, w) = \text{softmax}\big(\beta_{\text{assoc}} w(s_t, s)\big)$ calculates the weights for each stimuli s, including s_t itself. The β_{assoc} is the inverse temperature. Stimuli with a higher association gain more weight in estimating the state-action value for the current stimulus s_t. $T(s_t, a) = 1$ means the stimulus-action association has been trained before.

Like many other RL models (such as LC, ACL in this study), the MA model creates a policy by passing the state-action values through a softmax function,

$$\pi(a|s_t) = \text{softmax}\big(\beta Q_s(s_t, a)\big) \tag{29}$$

In total, the MA model has 4 parameters: the learning rate for the feature-action value function α_Q, the learning rate for the stimuli association matrix α_{assoc}, the inverse temperature for the association-based weights β_{assoc}, and the inverse temperature for the policy β.

4.4.9 ACL: attention at choice and learning

The ACL model is an algorithmic-level model that model humans' rewarding feature extraction ability using a linear selective attention mechanism. The model was from Leong et al., (2017) with two modifications. First, the original ACL model was developed on a different paradigm and could not be directly applied to the current generalization task. We modified the ACL model to include a feature-action value $Q(f, a)$ design as Ballard et al., (2018). Second, instead of using the attention weights calculated from the eye-tracking and functional MRI data, we estimated the attention weight using an attention model. The original authors constructed two types of

models to examine the bidirectional relationship between learning and attention. The "choice models" utilize attention data, collected through eye tracking and fMRI, to predict human behaviors. Conversely, the "attention models" use human behavioral data as input and predict the recorded attention data. In this study, we implemented the ACL model by combining the best choice model (the ACL model in the original paper) and the best attention model (the VALUE model in the original paper). We chose this approach because our study lacks attention data, such that we have to use the best attention model to provide reasonable estimation of the attention weights.

The action value of a stimulus s_t is a weighted sum of the feature-action value (Equation 1 in original paper),

$$Q(s_t, a_t) = \sum_{f \in \mathcal{F}(s_t)} \phi(f) Q(f, a_t) \tag{30}$$

where ϕ indicates the selective attention on each feature. The estimation of ϕ will be shown soon in Eq. 34. Note that, here we slightly abused the notation $\mathcal{F}(s_t)$. This feature embedding function in the ACL model is no longer returning a feature vector. Instead, it returns all features contains in a stimulus s. We did this to better align with the equations written in Leong et al., (2017). When receiving a reward r_t, the prediction error δ_t can be calculated as (Equation 2 in original paper),

$$\delta = r_t - Q(s_t, a_t) \tag{31}$$

which is then used to update the value of the chosen feature and action (Equation 3 in original paper),

$$Q(f, a_t) = Q(f, a_t) + \eta \phi(f) \delta_t, \forall f \in \mathcal{F}(s_t) \tag{32}$$

where η is the learning rate.

The ACL model generates actions by passing the stimulus-action value through a softmax function (Equation 4 in original paper).

$$\pi(a|s_t) = \mathrm{softmax}\big(\beta Q(s_t, a)\big) \tag{33}$$

where β is the inverse temperature.

To estimate the attention, the original paper used the VALUE model. This model presumes that attention tracks feature values, $\tilde{Q}(s, a)$. Note that this $\tilde{Q}(s, a)$ is a new value function used to estimate attention weight, independent from the Q function $Q(s, a)$ used in the choice model.

As described in original paper: "*The value of chosen features was updated based on the prediction error scaled by a subject-specific update rate, while the value of unchosen features was decayed toward 0 at a subject-specific decay rate* (p461, Leong et al., 2017)". We thereby updated this feature values as:

$$\tilde{Q}(f, a_t) = \tilde{Q}(f, a_t) + \eta_{\text{attn}}\delta_t, \forall f \in \mathcal{F}(s_t) \tag{34}$$

where $\delta_t = r_t - \tilde{Q}(s_t, a_t)$ and η_{attn} is the subject-specific update rate. This is very similar to Eq. 32, except for not modulated by the attention mechanism. For the unchosen features, we decayed their value towards 0,

$$\tilde{Q}(f, a_t) = \tilde{Q}(f, a_t) + \epsilon\left(0 - \tilde{Q}(f, a_t)\right), \forall f \notin \mathcal{F}(s_t) \tag{35}$$

where ϵ is the decay rate. Given that "*…the maximum feature value in each dimension was then passed through a softmax function to obtain the predicted attention vector…*(p461, Leong et al., 2017)", we defined the maximum feature value in each dimension, $d \in \{\text{shape}, \text{color}, \text{appendage}\}$:

$$\xi(d) = \max_{f \in d} V(f) \tag{36}$$

where $V(f) = \max_a Q(f, a)$, a popular method that has been widely used to approximate the value of a feature marginalized over actions (Sutton & Barto, 2018). Thus, the attention can be calculated as,

$$\phi(f) = \text{softmax}\big(\beta_{\text{attn}}\xi(d)\big), \qquad \forall f \in d \tag{37}$$

where β_{attn} is the inverse temperature.

The five parameters $\{\eta, \beta, \eta_{\text{attn}}, \epsilon, \beta_{\text{attn}}\}$ of the ACL are: the learning rate for the value η, the inverse temperature β, the learning rate for the feature value in attention estimation η_{attn}, the decay rate of the feature value in attention estimation ϵ, the inverse temperature in attention estimation β_{attn}.

4.4.10 L1PG, L2PG, and DCPG

The three models have similar computational goals with the ECPG model, except that the representation complexity $I^{\psi}(S; Z; \theta)$ terms in Eqs. 8 and 13 was respectively replaced by L1 norm ($||\theta||_1$), L2 norm ($||\theta||_2$), and decoder complexity $I^{\rho}(Z; A; \phi)$.

4.4.11 RNDPG: random regularizer policy gradient

The RNDPG considers a commonly used regularizer in machine learning: injecting noise into the encoder weights (Noh et al., 2017). Due to the injected noise $\varepsilon \in \mathbb{R}^{|F|\times|Z|}$ to the model, the likelihood calculation should be turned to an expectation form,

$$\log \pi(a_t|s_t,\xi) = \log E_{\varepsilon\sim p(\varepsilon)}[\pi(a_t,\varepsilon|s_t,\xi)] \tag{38}$$

Since our data lacks information regarding the type of noise the brain may have used in the current experimental trial t, the most effective approach is to account for this uncertainty by integrating over a presumed noise distribution . Accordingly, the update formula of the encoder parameter should be updated as,

$$\max_{\theta} J(\theta) = \max_{\theta} E_{\varepsilon\sim p(\varepsilon)}\left[\sum_{z}\psi(z|s_t;\theta,\varepsilon)[\rho(a_t|z)(r(s_t,a_t)-b)\ ]\right] \tag{39}$$

Assume the noise follows a Gaussian distribution $N(0,\lambda^2)$, we can use the reparameterization trick by first sampling ε from a standard Gaussian distribution $N(0,1)$ and multiple by λ,

$$\psi(z|s_t;\theta,\varepsilon) = \ \mathrm{softmax}(\theta(f_t,z)+\lambda\varepsilon) \tag{40}$$

Eq. 40 then has no closed-form solution, so we need to approximate it using the sampling method.

$$\max_{\theta} J(\theta) = \max_{\theta}\frac{1}{N_{\text{sample}}}\sum_{i=1}^{N_{\text{sample}}}\sum_{z}\psi(z|s_t;\theta,\varepsilon_i)[\rho(a_t|z)(r(s_t,a_t)-b)] \tag{41}$$

Where $N_{\text{sample}} = 50$. Similarly, the update formula for decoder parameters is,

$$\max_{\phi} J(\phi) = \max_{\phi}\frac{1}{N_{\text{sample}}}\sum_{i=1}^{N_{\text{sample}}}\sum_{z}\rho(a_t|z;\phi)[\psi(z|s_t,\varepsilon_i)(r(s_t,a_t)-b)] \tag{42}$$

Like other policy gradient model, the RNDPG model has three parameters $\{\alpha_\psi,\alpha_\rho,\lambda\}$: the learning rate for the encoder α_ψ, the learning rate decoder α_ρ, and the standard deviation of the injected noise λ.

4.5 Pretrain an encoder

AE describes the phenomenon where generalization between two “superficially dissimilar” stimuli increases after they have been paired with the same actions. To ensure their dissimilarity, we selected stimuli that are easily distinguishable by human participants. We operationally

defined this dissimilarity by setting a criterion: all four input stimuli must be classifiable with an accuracy of 99%. In order to accurately model human behaviors in the AE task, all models should undergo pretraining to achieve this level of discrimination accuracy.

The key step of the pretraining is to search for an appropriate θ_0. This is because all models' encoders were specially designed such that they can be initialized once the θ_0 is decided,

$$\bar{\psi}(z|s;\theta_0) = \begin{cases} \text{softmax}\big(\theta_0 \mathbf{I}(s,z)\big), & \text{for ECPG/CAPG} \\ \text{softmax}\big(\theta_0 \text{sim}(s,z)\big), & \text{for fECPG/fCAPG} \end{cases} \tag{43}$$

The initial discrimination accuracy of the encoder can be quantified as follows:

$$\text{acc}(\theta_0) = \frac{1}{4} \sum_{i \in \{x, x', y, y'\}} \bar{\psi}(z = i | s = i; \theta_0) \tag{44}$$

Given these constructs, we can search for an appropriate θ_0 by addressing the following objective,

$$\theta_0^* = \arg\min_{\theta_0} \frac{1}{2} \big(0.99 - \text{acc}(\theta_0)\big)^2 \tag{45}$$

Addressing this optimization objective, we initialized ECPG and CAPG model using $\theta_0^* = 5.232$. To initialize the fECPG and fCAPG, we used $\theta_0^* = 1.459$ for the consistent and conflict case; $\theta_0^* = 1.329$ for the control case.

4.6 Generate an action

A standard RL problem considers decision-making as sampling an action from the policy $\pi(a|s_t)$, a categorical distribution over the possible action space. In the AE problem, the possible action space varied from trial to trial. Participants were instructed to choose between a_1 and a_2 in one trial, while they were presented with a_3 and a_4 in another trial. To run both RL-base models in the AE task, we applied a technique called *invalid action masking* (Huang & Ontañón, 2020).

The simplest masking is to add a large negative number ζ (in this thesis, $\zeta = -1e12$) to logits of the actions that are not presented in the current trial. That is, when an RLPG agent needs to choose between a_1 and a_2, we can calculate its renormalized policy as,

$$\tilde{\pi}\big(a\big|s_t;\tilde{\phi}\big) = \text{softmax}\left(\tilde{\phi}(s_t,a)\right) \tag{46}$$

where $\tilde{\phi}$ is,

$$\tilde{\phi}(s,a) = \begin{cases} \phi(s,a), & \text{if } a \text{ is valid action at } t \\ \phi(s,a) + \zeta, & \text{if } a \text{ is invalid action at } t \end{cases} \quad (47)$$

We then sampled from this renormalized policy to model human decision-making.

For the ECPG and CAPG models, we masked and re-normalized the decoder for all representations z,

$$\tilde{\rho}\left(a \middle| z; \tilde{\phi}\right) = \text{softmax}\left(\tilde{\phi}(z,a)\right) \quad (48)$$

4.7 Perturbation-based feature importance

To investigate the importance of different features (color, shape, appendages) to humans, we adopted a *perturbation-based* measurement approach (Greydanus et al., 2018; Guo et al., 2021), applied to our model fitted to human behavior. This method calculates the importance of each feature by theorizing that if an agent focuses heavily on a particular feature, then a minor perturbation in that feature might lead to significant changes in the output. This perturbation-based importance has been applied to extract measures of attention from large-scale deep reinforcement learning models in artificial intelligence, which was shown to be similar to human eye-tracking attention data (Guo et al., 2021).

In the present work, we calculated the perturbation-based feature importance following the pseudo-algorithm 1 (**Extended Data 5.4**), modified based on the feature importance algorithm described in (Fisher et al., 2019).

4.8 Model fitting

For each model, we estimated its free parameters separately for each subject, using all behavioral data from both the training and testing trials without cross-validation. This approach is consistent with many previous human learning studies, which are often structured with 2 to 4 parallel blocks due to various practical constraints (e.g. Browning et al., 2015; Daw et al., 2011; Gershman, 2020; Rac-Lubashevsky et al., 2023). Given the insufficient number of blocks, these studies, including ours, do not meet the prerequisites for effective cross-validation.

The parameters were estimated via maximum a posteriori (MAP):

$$\max_{\xi} \sum_{i=1}^{N} \log \pi(a_i | s_i, \mathrm{M}, \xi) + \log p(\xi) \tag{49}$$

where M refers to the model architecture, and ξ the model parameters. N is the number of trials for each participant. s_i and a_i are the presented stimuli and human responses recorded on each trial. We selected a very flat prior $p(\xi) = \mathrm{Halfnorm}(0, 50)$ for all parameters with a range of $(0, \infty)$ only to avoid extreme parameter values without biasing estimation. This prior is uninformative yet ensures that parameter estimates remain within a reasonable range. Parameters with a range of $(0, 1)$ used a uniform prior.

Parameter estimation was performed using the BFGS algorithm, implemented with the Python package `scipy.minimize`. For each participant, we ran the algorithm with 50 different randomly chosen parameter initializations to avoid local minima in the non-convex landscape.

4.9 Simulation

The parameter we are interested in is the simplicity degree (λ). We simulated the ECPG model's learning and generalization behaviors by varying λ, while keeping the other two learning rate parameters constant. In Experiment 1, the learning rate of the encoder was fixed at $\alpha_\psi = 40$, and that of the decoder was fix at $\alpha_\rho = 4$. In Experiment 2, the learning rate of the encoder was fixed at $\alpha_\psi = 8$, and that of the decoder was fix at $\alpha_\rho = 4$.

4.10 Correlation between humans' and models' probe response

For each participants within a block, we calculated the frequency for each action as an estimation of human probe policy. We applied the same method to the simulated data to obtain models' probe policy. Subsequently, we computed the Spearman's correlation between human participants and models based on the probability of selecting actions a_1 and a_3. These two actions sufficiently characterize a policy.

5. Extended Data

5.1 Model parameters

Table S1: model parameters in Experiment 1 (mean±standard deviation)

RLPG	$\alpha_\pi \in (0, \infty)$		
	0.567±0.630		
CAPG	$\alpha_\psi \in (0, \infty)$	$\alpha_\rho \in (0, \infty)$	
	4.376±6.519	1.553±1.753	
ECPG	$\alpha_\psi \in (0, \infty)$	$\alpha_\rho \in (0, \infty)$	$\lambda \in (0, \infty)$
	14.983±14.881	2.336±2.926	0.112±0.277

The reported values are from the 2.5% to 97.5% quantiles with extreme values removed.

Table S2: model parameters in Experiment 2 (mean±standard deviation)

fRLPG	$\alpha_\pi \in (0, \infty)$				
	0.567±0.630				
fCAPG	$\alpha_\psi \in (0, \infty)$	$\alpha_\rho \in (0, \infty)$			
	4.376±6.519	1.553±1.753			
fECPG	$\alpha_\psi \in (0, \infty)$	$\alpha_\rho \in (0, \infty)$	$\lambda \in (0, \infty)$		
	14.983±14.881	2.336±2.926	0.112±0.277		
LC	$\eta \in (0, 1)$	$\alpha \in (0, \infty)$	$\beta \in (0, \infty)$		
	0.108±0.232	1.637±5.515	127.218±416.861		
MA	$\alpha_Q \in (0, 1)$	$\alpha_{\text{assoc}} \in (0, 1)$	$\beta_{\text{assoc}} \in (0, \infty)$	$\beta \in (0, \infty)$	
	0.090±0.117	0.730±0.379	7.481±4.456	8.977±8.106	
ACL	$\eta \in (0, 1)$	$\beta \in (0, \infty)$	$\eta_{\text{attn}} \in (0, 1)$	$\epsilon \in (0, 1)$	$\beta_{\text{attn}} \in (0, \infty)$
	0.393±0.357	7.184±6.221	0.609±0.362	0.217±0.343	25.452±17.876
fL1PG	$\alpha_\psi \in (0, \infty)$	$\alpha_\rho \in (0, \infty)$	$\lambda \in (0, \infty)$		
	1.016±2.191	1.779±1.396	0.056±0.246		
fL2PG	$\alpha_\psi \in (0, \infty)$	$\alpha_\rho \in (0, \infty)$	$\lambda \in (0, \infty)$		
	1.479±2.279	1.579±1.185	0.107±0.671		
fRNDPG	$\alpha_\psi \in (0, \infty)$	$\alpha_\rho \in (0, \infty)$	$\lambda \in (0, \infty)$		
	3.390±2.837	2.617±2.373	1.181±0.879		
fDCPG	$\alpha_\psi \in (0, \infty)$	$\alpha_\rho \in (0, \infty)$	$\lambda \in (0, \infty)$		

	30.620±43.514	0.027±0.095	0.012±0.038		

The reported values are from the 2.5% to 97.5% quantiles with extreme values removed

5.2 A graphical illustration of the main models

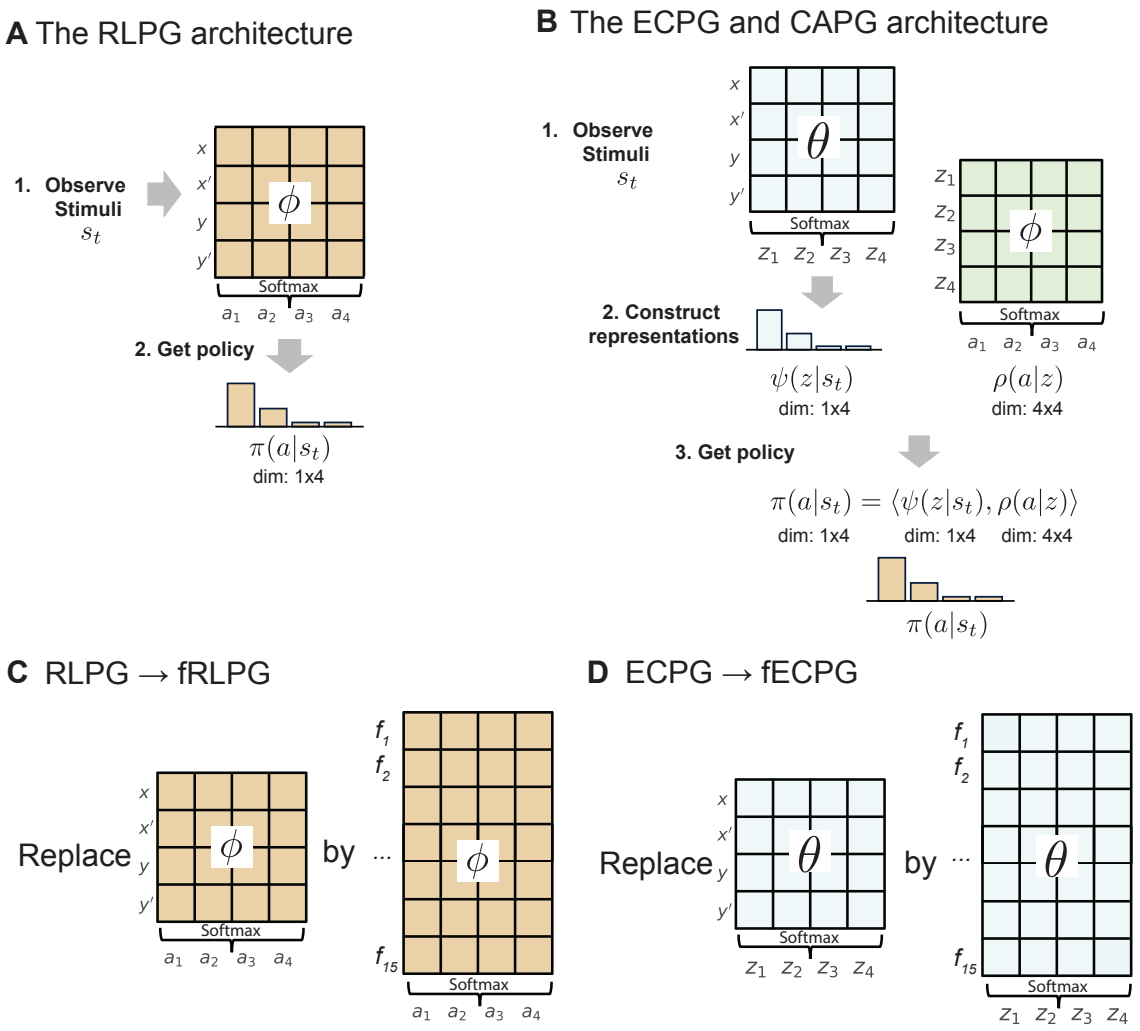

Figure S1 The semantic of the main model. $\langle \cdot, \cdot \rangle$ means dot product.

A. For each trial, the RLPG model observes a stimuli s_t and index the corresponding policy $\pi(a|s_t)$ for response.

B. The ECPG and the CAPG model needs to reconstruct the observed stimuli as an internal representation $\psi(z|s_t)$ and use the representation to derive a policy $\pi(a|s_t)$.

C. To build a feature-based RLPG, we only need to replace the 4(stimuli)-by-4(actions) policy with a 15 (features) by 4 (actions) policy.

D. To build a feature-based ECPG or CAPG, we only need to replace the 4(stimuli)-by-4(representations) encoder with a 15(features)-by-4(representations) encoder.

5.3 Representation evolution for all λs

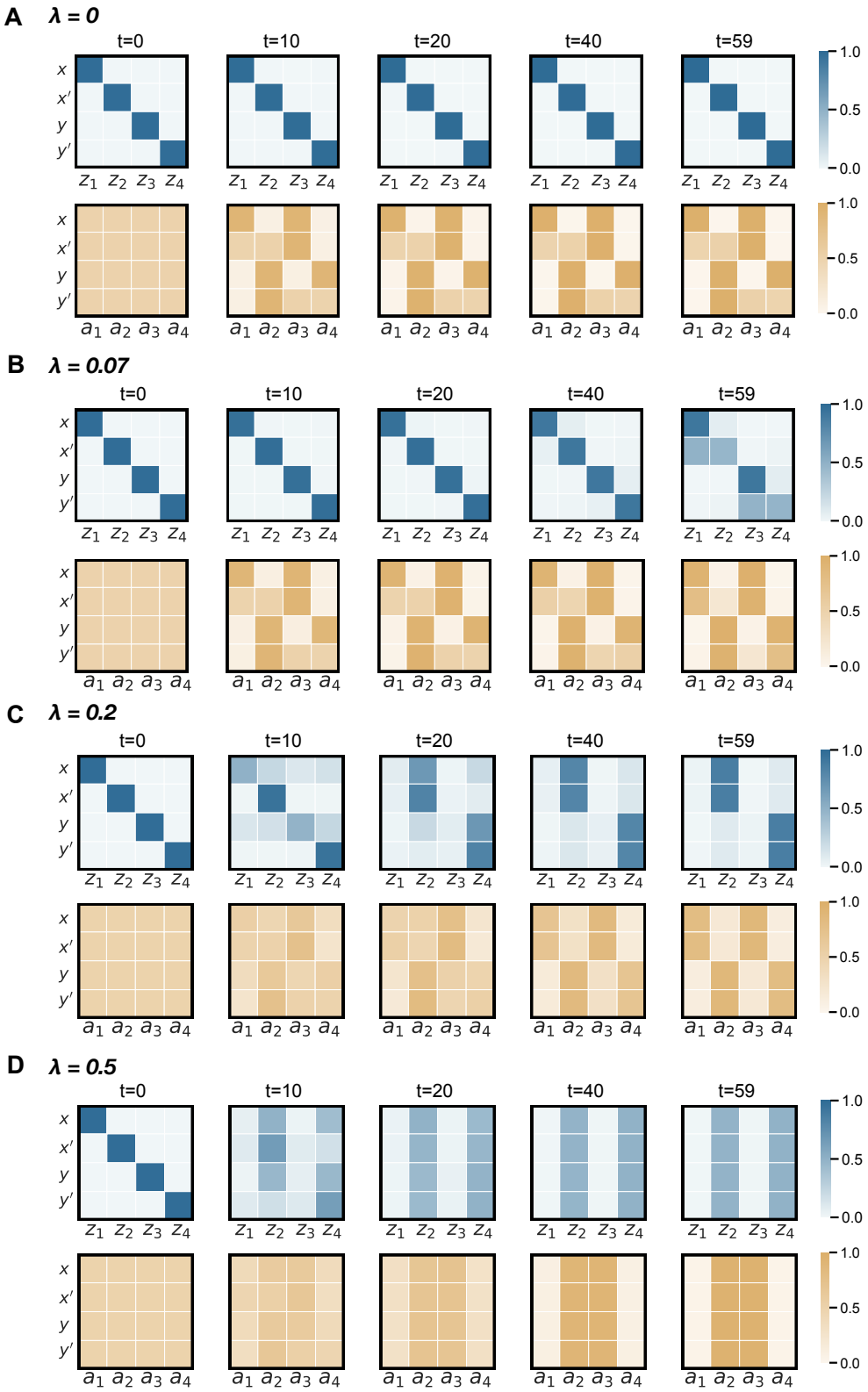

Figure S2 Encoders and Policies for ECPG model with different simplicity tendency, λ.

5.4 Pseudo algorithm for perturbation-based feature importance

Algorithm 1 Calculate the perturbation-based feature importance of an encoder.

Require: Encoder ψ
Require: Feature embedding function $\mathcal{F}$
Require: Stimulus space $\{x, x', y, y'\}$
Require: Feature space $\{\text{shape}, \text{color}, \text{appendage}\}$
Require: Number of sample M
  for $d \in \{\text{shape}, \text{color}, \text{appendage}\}$ **do**
    for $s \in \{x, x', y, y'\}$ **do**
      $f_{\text{orig}} = \mathcal{F}(s)$ ▷ Embed the stimulus
      $z_{\text{orig}} = \psi(f_{\text{orig}})$ ▷ Encode the original stimulus
      for i=1:N **do**
        f^i_{pert} = change the one-hot code of feature d to another value
        $z^i_{\text{pert}} = \mathcal{F}(f^i_{\text{pert}})$ ▷ Encode the perturbed stimulus
        $k^i = D_{\text{KL}}(z_{\text{orig}} \mid z_{\text{pert}})$ ▷ Compute the KL divergence
      end for
      $\text{imp}(s) = 1/N \sum_i^N k^i$ ▷ Importance, per stimulus per dimension
    end for
    $\text{IMPORTANT}(d) = 1/4 \sum_s \text{imp}(s)$ ▷ Importance, per dimension
  end for
  Return IMPORTANT

5.5 Model fitting in Experiment 1 for all models

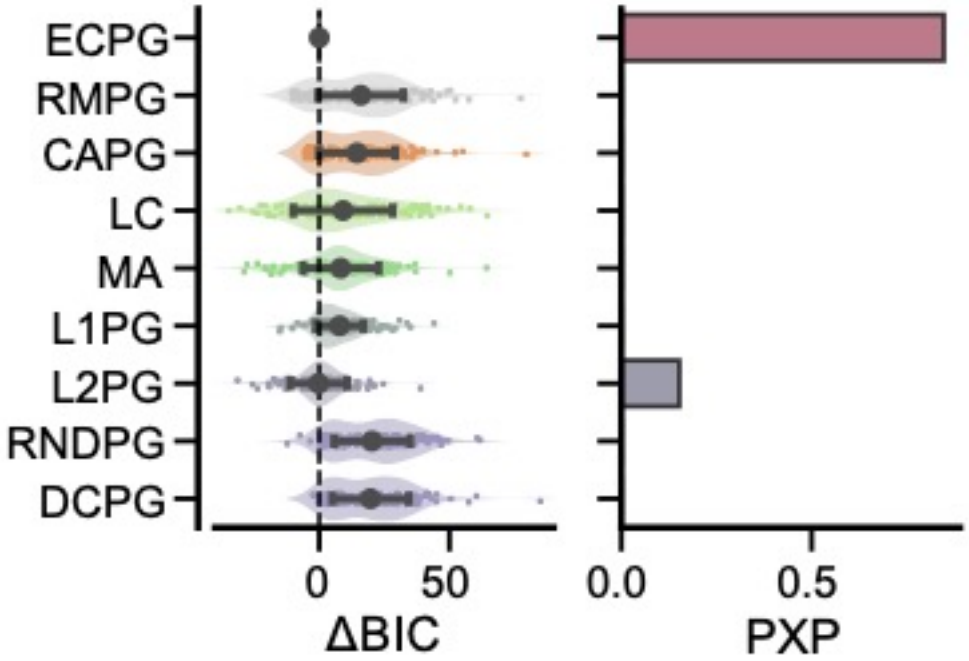

Figure S3 Model fitting performance for all candidate models in Experiment1. Error bars represent standard deviations. The ACL model, which must independently learn the value of each perceptual feature, was not applicable for Experiment 1. The L2PG model exhibits very close performance with our main ECPG model in terms of BIC. However, the PXP still significantly favors the ECPG model over the L2PG model (85% vs. 15%). This indicates that while the L2PG model is good at capturing certain behavioral patterns that are also discernible by other models, the ECPG model has a unique capability to identify specific behavioral patterns that other models fail to identify. See **Supplemental Note 2.3** for more details on the on the comparison between ECPG and L2PG. ECPG is the best model across the two datasets.

5.6 Learning and generalization performance in the testing stage of Experiment 2

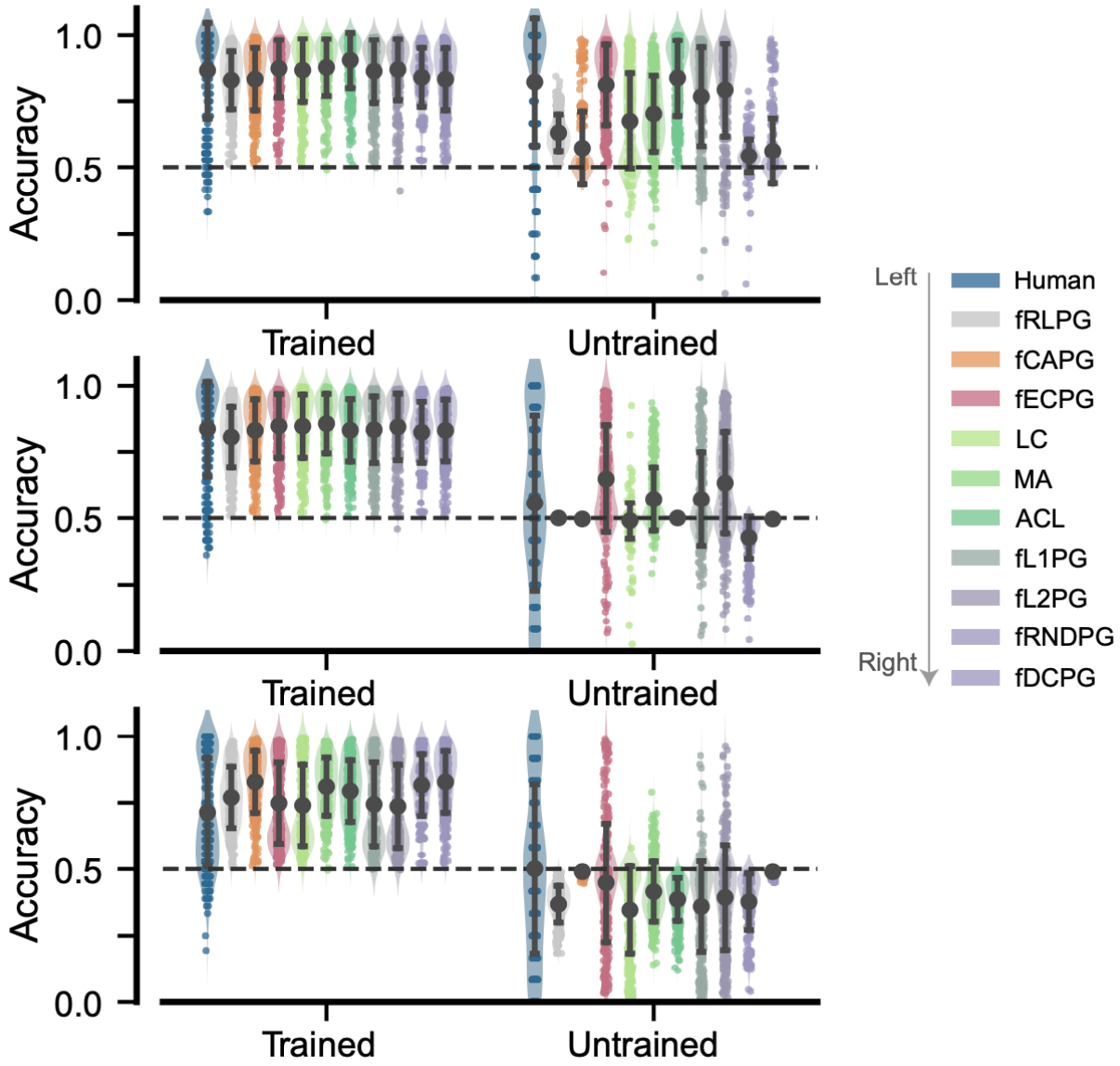

Figure S4 Learning and generalization performance in the testing stage for all candidate models. Standard deviations are denoted by error bars. The models fRLPG, fCAPG, ACL, and fDCPG demonstrate a failure to generalize in the testing stage, suggesting a deficiency in their capacity for state abstraction. The two algorithmic-level models, LC and MA, which incorporate handcrafted state abstraction, show some degree of state abstraction capabilities but still fall short of human performance. In contrast, the fECPG model and its heuristic approximations, fL1PG and fL2PG, which aim to minimize representational complexity, effectively capture human learning and generalization capabilities.

5.7 Response to the probe stimuli

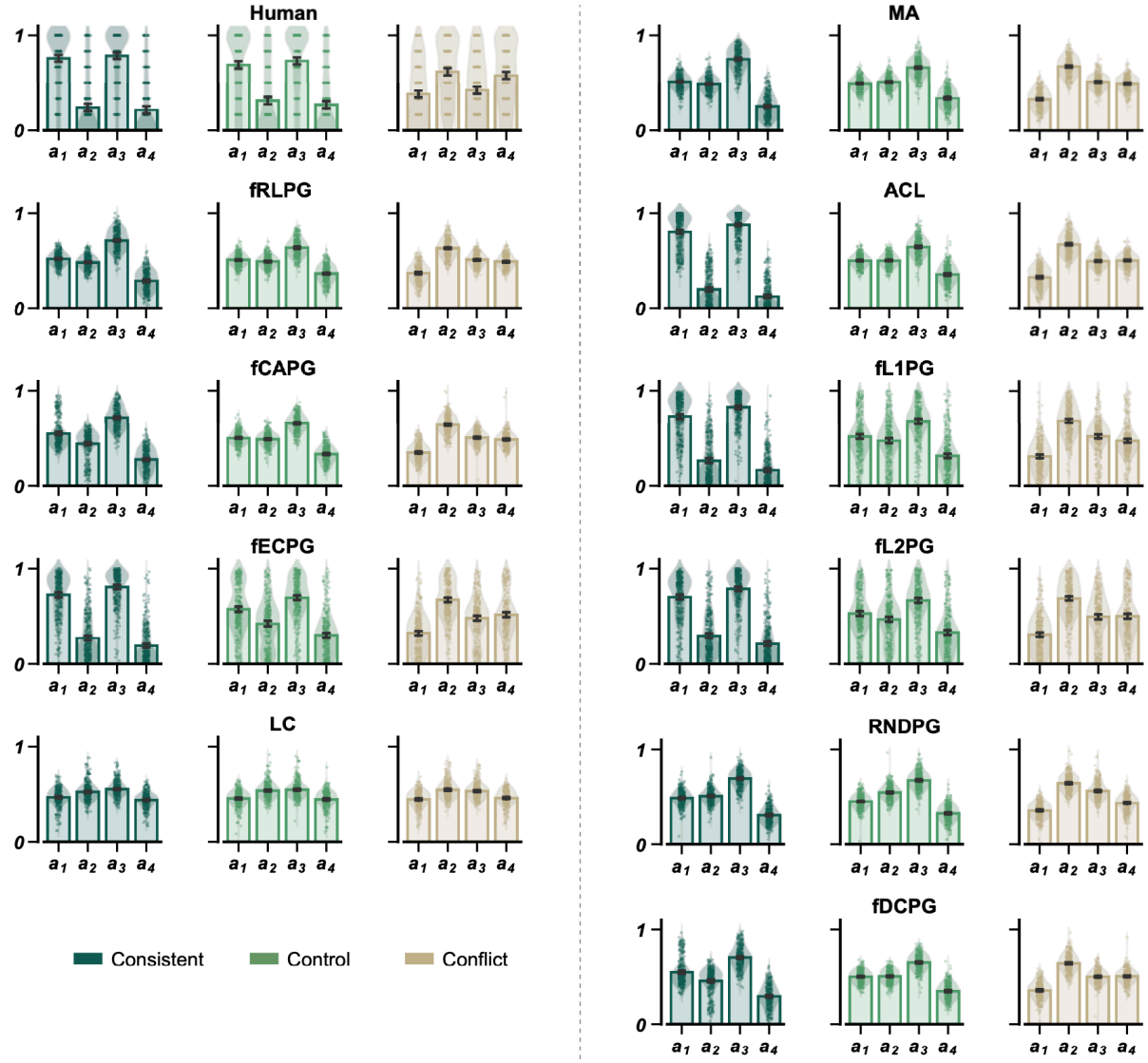

Figure S5: Responses to the probe stimuli for all candidate models. Error bars stand for a 95% confidence interval. The models fRLPG, fCAPG, LC, MA, and fDCPG cannot explain human probe responses, suggesting a deficiency in their capacity for rewarding feature extraction. The ACL model, equipped with a built-in feature extraction mechanism, only succeeds in consistent cases and fails in the other two. Finally, the fECPG model and its heuristic approximations, fL1PG and fL2PG, which aim to minimize representational complexity, successfully account for human response to the probe stimuli.

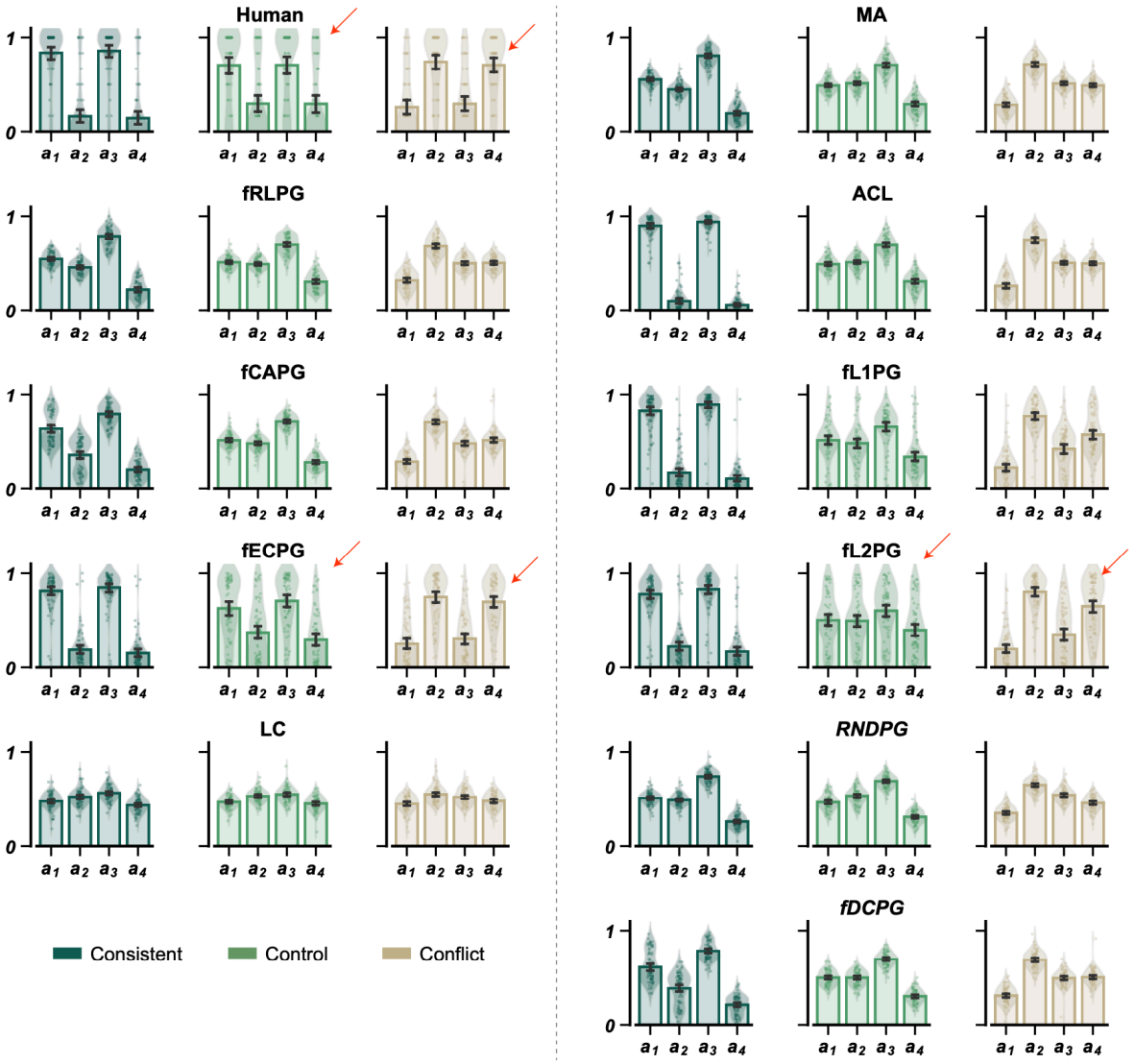

Figure S6: Responses to the probe stimuli for all candidate models for participants with better generalization. Out of 313 valid participants, we specifically selected 79 who has the top 25% for untrained accuracy to form a “high generalization” group. The fECPG model demonstrated a significantly higher fitting advantage over all other models, including the fL2PG. The red arrows highlight that the ECPG model's alignment with human data is more precise in both control and conflict scenarios

5.8 The model recovery result

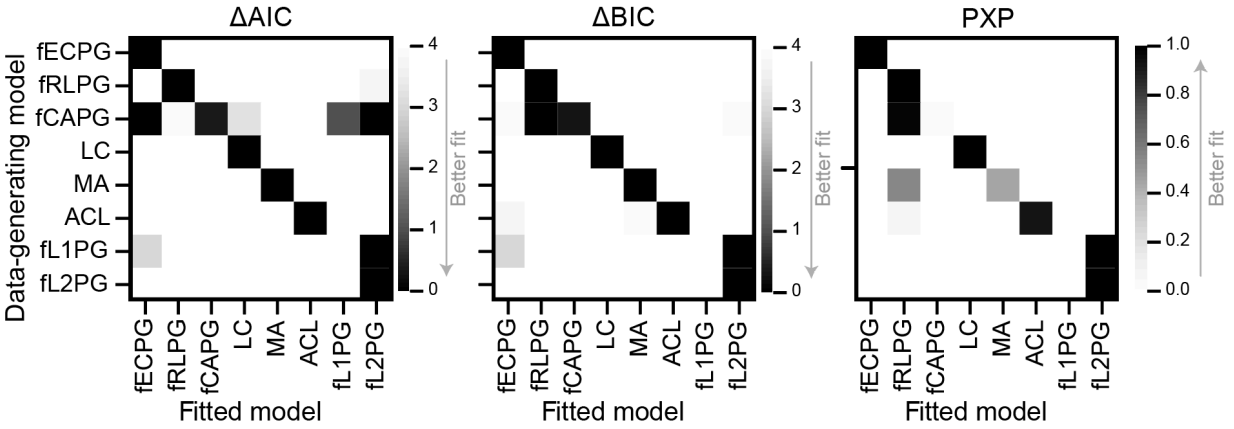

Figure S6: Model recovery results for Experiment 2. The AIC and BIC values are presented as relative metrics, computed by subtracting the values from the best-fitting model (across columns). The PXP metric is critical in the model recovery analysis. We sampled 40 participants and used their fitted parameters to generate 10 simulated runs of behavioral data for Experiment 2, with three blocks per run, each representing one experimental condition. This resulted in 8 (models) × 40 (parameter sets) × 10 (samples) = 3,200 synthetic datasets. We fitted all models to these synthetic datasets using the same fitting method as described previously, except for the RNGPG and DCPG models due to their poor fitting performance. To control for randomness, we averaged the fitting results (AICs and BICs) within each parameter set over the 10 sample runs. The model recovery analysis reveals three key observations: first, the ECPG model can be uniquely distinguished from the other models and does not falsely fit well to simulated data from other models, demonstrating that its superior performance is due to its accurate description of human behavior rather than greater expressiveness; second, based on BIC and PXP, the CAPG model was always recovered as the RLPG model. This further verifies that the success of the ECPG is not due to its cascade architecture but to its efficient coding computational model; and third, the two norm-based regularizers (L1 and L2) cannot be accurately differentiated from each other but can be separated from the ECPG model, supporting our claim that the ECPG and the L2PG models capture different aspects of human behavior. Overall, these findings indicate that our model recovery approach is sufficient to confirm our conclusion that the ECPG model, with its augmented RL objective incorporating efficient coding, best accounts for human learning and decision-making.