How to talk so AI will learn: Instructions, descriptions, and autonomy

Theodore R. Sumers
Computer Science
Princeton University
sumers@princeton.edu

Robert D. Hawkins
Princeton Neuroscience Institute
Princeton University
rdhawkins@princeton.edu

Mark K. Ho
Computer Science
Princeton University
mho@princeton.edu

Thomas L. Griffiths
Computer Science, Psychology
Princeton University
tomg@princeton.edu

Dylan Hadfield-Menell
EECS, CSAIL
MIT
dhm@csail.mit.edu

Abstract

From the earliest years of our lives, humans use language to express our beliefs and desires. Being able to talk to artificial agents about our preferences would thus fulfill a central goal of value alignment. Yet today, we lack computational models explaining such language use. To address this challenge, we formalize learning from language in a contextual bandit setting and ask how a human might communicate preferences over behaviors. We study two distinct types of language: *instructions*, which provide information about the desired policy, and *descriptions*, which provide information about the reward function. We show that the agent's degree of autonomy determines which form of language is optimal: instructions are better in low-autonomy settings, but descriptions are better when the agent will need to act independently. We then define a pragmatic listener agent that robustly infers the speaker's reward function by reasoning about *how* the speaker expresses themselves. We validate our models with a behavioral experiment, demonstrating that (1) our speaker model predicts human behavior, and (2) our pragmatic listener successfully recovers humans' reward functions. Finally, we show that this form of social learning can integrate with and reduce regret in traditional reinforcement learning. We hope these insights facilitate a shift from developing agents that *obey* language to agents that *learn* from it.

1 Introduction

As artificial agents proliferate in society, aligning them with human values is increasingly important [1–3]. But how can we build machines that understand what we want? Prior work has highlighted the difficulty of specifying our desires via numerical reward functions [3–5]. Here, we explore language as a means to communicate them. While most previous work on language input to AI systems focuses on *instructions* [6–19], we study instructions alongside more abstract, *descriptive* language [20–25]. We examine how humans communicate about rewards, and formalize learning from this input.

To consider how humans communicate about reward functions, imagine taking up mushroom foraging. How would you learn the rewards associated with different fungi (i.e. which are delicious and which are deadly)? In such a setting, learning from direct experience [26] is risky; most humans would seek to learn *socially* instead. So how might we learn reward functions from others? Prior work in reinforcement learning (RL) has examined a number of social learning strategies, including passive *inverse reinforcement learning* (observe an expert pick mushrooms, then infer their reward

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

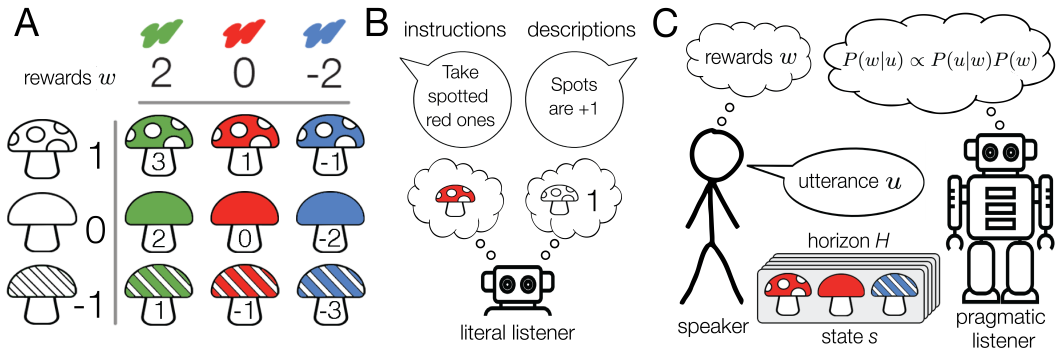

Figure 1: We formalize learning from language in a contextual bandit setting. **A**: Linear rewards associated with features determine whether actions (mushrooms) are high or low reward. **B**: We consider two forms of language. *Instructions* tell the listener to take a specific action, while *descriptions* provide information about underlying reward function. A *literal* listener interprets utterances according to these fixed semantics. **C**: In this work, we introduce a speaker model suggesting that humans use instructions and descriptions adaptively to maximize a literal listener's expected rewards over some *horizon*, which formalizes the agent's degree of autonomy. We then define a *pragmatic* listener which uses Bayesian inference to recover the speaker's latent reward function.

function [27, 28]) or active *preference learning* (offer an expert pairs of mushrooms, observe which one they eat, and infer their reward function [29–31]).

However, few humans would rely on such indirect data if they had access to a cooperative teacher [32–35]. For example, an expert guiding a foraging trip might *demonstrate* [36, 37] or verbally *instruct* [7] the learner to pick certain mushrooms, licensing stronger inferences. While such pedagogical actions have been useful for guiding RL agents [10, 17–19], natural language affords richer, under-explored forms of teaching. For example, an expert teaching a seminar might instead *describe* how to recognize edible or toxic mushrooms based on their features, thus providing highly generalized information.

To formalize this process, we present a model of learning from language in a contextual bandit setting (Fig 1). We propose a speaker model that chooses utterances to maximize the listener's expected rewards over some task *horizon*. The horizon quantifies notions of autonomy described in previous work [13, 14, 25]. If the horizon is short, the agent is closely supervised; at longer horizons, they are expected to act more independently. We then analyze *instructions* (which provide a partial policy) and *descriptions* (which provide partial information about the reward function). We show that instructions are optimal at short horizons, while descriptions are optimal at longer ones.

Second, we consider how a listener might learn from such a speaker. We define a pragmatic listener that infers the speaker's latent reward function based on their utterances. While prior work suggests pragmatic learning can be vulnerable to model mis-specification [38], we show that jointly inferring the speaker's horizon and reward function can mitigate this risk.

Finally, we conduct an online behavioral experiment which shows our models support strong reward inference. We integrate this social inference with traditional RL and find that it accelerates learning and reduces regret. Overall, our results suggest that descriptive language and pragmatic inference are powerful mechanisms for value alignment and learning.[1]

2 Related work

Classic RL assumes that the reward function is given to the agent [26]. However, in practice, it is difficult to specify a reward function to obtain desired behavior [3], motivating *learning* the reward function from social input.[2] Most social learning methods assume the expert is simply acting optimally [27, 28, 42], but recent *pragmatic* methods [36, 43, 44] instead assume the expert is actively teaching.

[1] Code and data are available at `https://github.com/tsumers/how-to-talk`.

[2] Another approach bypasses learning the reward function and allows the human to provide a reward signal directly [39, 40]. This is challenging, however, as people do not typically provide classical reinforcement [41].

Learning reward functions from observed actions. When the desired behavior is known (but the reward function is not), inverse reinforcement learning [IRL, 13, 14, 27, 28, 42, 45, 46] can be used to infer an expert's reward function from their actions. However, such approaches face fundamental issues with identifiability: observed behavior can often be explained by multiple reward functions. One solution allows the agent to actively query the human [30, 31, 47–49]. An alternative is to make a stronger assumption: that the human is actively teaching [i.e. behaving *pedagogically*, 36, 43, 44]. We next review work on learning rewards from language, then return to these methods.

Learning reward functions from instructions. Instructions use language to communicate specific actions or goals [6–8]. Prior work has used instructions to shape reward functions [15–17] or learn a language-conditioned reward model [9–12]. An alternative approach uses instructions for value alignment. This paradigm assumes instructions reflect underlying preferences and uses them to infer this latent reward function [13, 14]. Our work extends this to incorporate other forms of language: we ask when a human would use an instruction (vs. other forms of language), and—given an instruction (vs. other forms of language)—what an agent should infer about the speaker's reward function.

Learning reward functions from descriptions. Rather than expressing specific goals, reward-descriptive language encodes abstract information about preferences or the world. The education literature suggests such rich feedback is crucial for human learning [50, 51]. However, a relatively smaller body of work uses it for RL: by learning reward functions directly from existing bodies of text [20–23] or interactive, free-form language input [24, 25]. Our work provides a formal model of such language in order to compare it with more typically studied instructions. Other related lines of work use language which describes agent *behaviors*. This language, whether externally provided [52–59] or internally generated [60–63], is typically used to structure the task representation or guide exploration. In contrast, we study language describing task-relevant properties of the environment.

Learning reward functions from pedagogy. The preceding algorithms all assume that training examples (whether demonstrations or language) are generated by a human that is *indifferent to the learning process*. Recent work has begun to challenge this assumption by considering *pedagogical* settings [34–37]. In particular, the Rational Speech Act framework [64, 65] builds on classic Gricean theory [66] to formulate optimal communication in terms of recursive Bayesian inference and decision-making. These ideas have been applied to a variety of language tasks including reference games [67, 68], captioning [69], and instruction following [70]. Developing analogues of linguistic pragmatics in reinforcement learning — i.e., algorithms that assume data are *intentionally designed to be informative* — is an active area of research [36, 38, 43, 44]. In particular, inverse reward design (IRD, [4]) uses pragmatic inference on numerical reward functions: rather than take the provided reward function literally, IRD quantifies uncertainty over the function to mitigate alignment risk. Similarly, we apply pragmatic inference to reward-related language. Applying IRD principles to reward-related language offers two primary benefits over its non-linguistic formulation. First, language is *accessible* for humans. While traditional IRD is useful when offline training RL agents, inferring rewards from language would be broadly useful for real-world, real-time interactions with non-experts. Second, language can address *future settings*: speakers can refer to actions or features which are not physically present. Speakers can thus provide information about rare or hazardous possibilities (e.g. poisonous mushrooms) which lie outside the listener's experience. We next describe a model for such language use.

3 Background

In this work, we consider the problem of *learning from language* in a *contextual bandit* setting. This section introduces models of language use from cognitive science, then describes contextual bandits.

The Rational Speech Acts framework (RSA). Our theoretical approach is based on the Rational Speech Acts framework (RSA, [64, 65, 71–73]), a computational model of language understanding. RSA uses theory-of-mind reasoning to explain how human listeners derive meaning from words. It begins by defining a *literal listener* L_0 which considers only the conventional (e.g., dictionary) definitions of words. These conventions are represented by a "lexicon" function $\mathcal{L}$ which maps from utterances to world states w.[3] RSA then assumes that a *speaker* S_1 makes a noisy-optimal

[3]The lexicon can be seen as the *grounding* of language [74, 75]. Most RSA models, including this work, assume that groundings are mutually known. However, methods have been developed to learn them [24, 25] or allow for uncertainty [76, 77], and we view integrating the problem of grounding as important future work.

selection from a finite set of utterances, choosing u according to some utility function $U(\cdot)$. Finally, a *pragmatic listener* L_1 infers the intended meaning by inverting the speaker model:

$$P_{L_0}(w \mid u) \propto \mathcal{L}(u, w) \tag{1}$$

$$P_{S_1}(u \mid w) \propto \exp\{\beta_{S_1} \cdot U(u, w)\} \tag{2}$$

$$P_{L_1}(w \mid u) \propto P_{S_1}(u \mid w) P(w) \tag{3}$$

In Eq. 2, $\beta_{S_1} > 0$ is an inverse temperature parameter. RSA models typically assume the speaker's objective is *informativeness*. Formally, the speaker tries to reduce the listener's information-theoretic uncertainty [78] over the true state of the world w^*: $U(u, w^*) = \ln P_{L_0}(w^* \mid u)$. Explicitly modeling the speaker as making a rational choice from a set of possible utterances allows the pragmatic listener to enrich the literal meaning of the utterance, inferring additional information about the world.

To take an intuitive example, we consider the classic phenomenon of *scalar implicature* [79–81]. If Alice announces "I ate some of the cookies," a typical listener Bob understands this as "some *but not all*." RSA explains this as follows. First, the word "some" *literally* maps to any world state where Alice ate more than zero cookies (e.g. 1, 2, 3... or all), while "all" maps only to the world state where she ate all of them (Eq. 1). Then, S_1 Alice is presumed to choose a maximally-informative utterance (i.e. one that specifies the world state as precisely as possible; Eq. 2). Finally, L_1 Bob reasons as follows: if Alice had eaten all of the cookies, saying "I ate *all* of the cookies" would have been more precise and thus preferred. Because she chose *not* to say this, he can *infer* that her statement means "some *but not all*" (Eq. 3). In this work, we define a pragmatic L_1 listener which uses RSA to infer the speaker's reward function. Crucially, such inferences are based on theory-of-mind and thus vulnerable to *misspecification* [38]: if L_1 Bob's model of S_1 Alice is wrong, his inferences will be wrong. We show how a carefully specified speaker model, incorporating both instructions and descriptions, allows our pragmatic listener to robustly infer a speaker's reward function.

Contextual bandits. Contextual bandits can be seen as one-step Markov Decision Processes [26, 82], thus isolating the problem of *learning* a reward function from the problem of *planning* sequential actions to maximize it. As such, they are a popular testbed for RL algorithms [83–87]. Formally, we define a set of A possible actions. Actions are associated with a binary feature vector $\phi : A \to \{0, 1\}^K$ (e.g., a mushroom may be green (or not), or striped (or not)). Following other work in IRL [4, 27, 42], we assume rewards are a linear function of these features (e.g., green mushrooms tend to be tasty):

$$R(a, w) = w^\top \phi(a) \tag{4}$$

so w is a vector that defines the value of each feature (see Fig. 1A). Each task consists of a sequence of H i.i.d. states. We refer to H as the *horizon.* At each time step $t < H$, the agent is presented with a state s_t consisting of a subset of possible actions: $s_t \subseteq A$ (e.g., a patch containing a set of mushrooms). They choose an action $a \in s_t$ according to their policy, $\pi_L : S \to \Delta(A)$.

4 Formalizing learning from language in contextual bandits

While bandits are typically considered as an *individual* learning problem, we instead ask how an agent should learn *socially* from a cooperative, knowledgeable partner. We formalize this by introducing a second agent: a speaker who knows the true rewards w and the initial state s_0, and produces an utterance u (Fig. 1C). This utterance may affect the listener's policy; with a slight abuse of notation, we denote this updated policy as $\pi_L(a \mid u, s)$. The listener then uses this policy to choose actions. The horizon H determines how many actions the listener will perform under this updated policy. Intuitively, $H = 1$ represents minimal autonomy (or maximal supervision, i.e. guided foraging), whereas $H \to \infty$ is maximal autonomy (or minimal supervision, i.e. teaching the listener to forage independently). We first assume H is known to both listener and speaker, then relax this assumption. This framework exposes two interrelated problems. First, what should a helpful speaker say? And second, how should the listener update their policy in light of this information?

4.1 Speaker model

We apply the RSA framework (Section 3) to a RL setting by introducing a new speaker objective: instead of Gricean informativeness [65, 66] we assume that speakers seek to *maximize the listener's rewards* [88]. Conceptually, this changes RSA's scope from epistemic to decision-theoretic. Instead of reasoning about how utterances affect the listener's *beliefs*, such a speaker reasons about how they

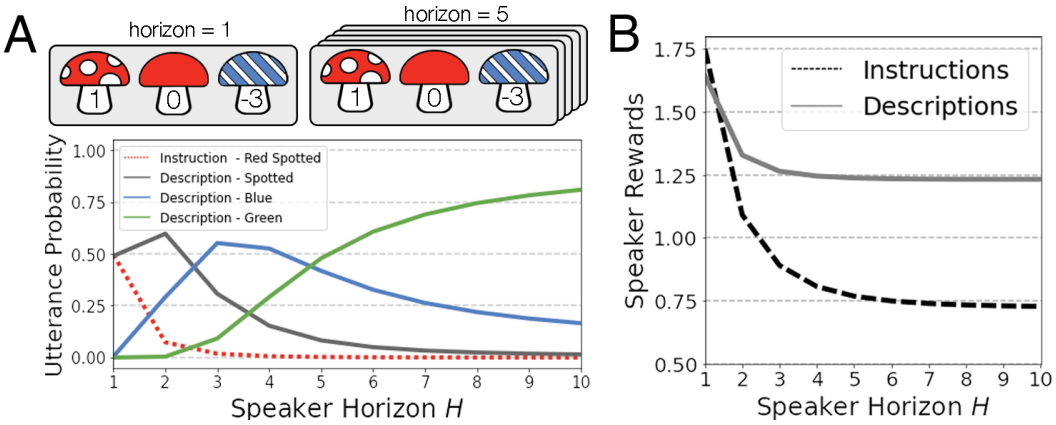

Figure 2: The *horizon* formalizes the agent's degree of autonomy and strongly affects a reward-maximizing speaker's choice of utterance. **A**: Top: as the horizon lengthens, the agent will act independently in more unknown future states. Bottom: simulating a reward-maximizing speaker. At short horizons, the speaker is focused on the spotted red mushroom. As the horizon lengthens, they are less biased towards this particular state and provide more generally useful information about the reward function. **B**: Speaker rewards (Eq. 7, averaged over all 84 start states) for a speaker with access to only instructions or only descriptions. Instructions outperform descriptions at short horizons, but descriptions offer much stronger generalization.

affect the listener's *policy*. When the state is known, we define the *present* utility of an utterance as the expected reward from using the updated policy to choose an action:

$$U_{\text{Present}}(u \mid s, w) = \sum_{a \in s} \pi_L(a \mid u, s) R(a, w) \tag{5}$$

However, we are interested in settings where agents may need to act autonomously [13, 24, 25]. Thus, we must consider how well the policy generalizes to unknown, future states. The *future* utility of an utterance with respect to a distribution over states $P(s)$ can be written as:

$$U_{\text{Future}}(u \mid w) = \sum_{s \in S} U_{\text{Present}}(u \mid s, w) P(s) \tag{6}$$

Because states are i.i.d. in the bandit setting, a speaker optimizing for a horizon H can be defined as a linear combination of Eqs. 5 and 6:

$$U_{S_1}(u \mid w, s, H) = \frac{U_{\text{Present}} + (H-1)U_{\text{Future}}}{H} = \frac{1}{H} U_{\text{Present}} + (1 - \frac{1}{H}) U_{\text{Future}} \tag{7}$$

where $H = 1$ reduces to Eq. 5 and as $H \to \infty$ reduces to Eq. 6. At short horizons, speakers are biased towards producing utterances that optimize the listener's policy in the present state. As the horizon lengthens, the speaker expects the listener to behave more autonomously. Generalization becomes more important, and this bias is reduced.

4.2 Formalizing instructions

We now consider how utterances affect the listener's policy. *Instructions* map to specific actions or trajectories [7, 13]; in our work, "instruction" utterances correspond to the nine actions (Fig 1A,B). Given an instruction, a literal listener executes the corresponding action. If the action is not available, the listener acts randomly:

$$\pi_{L_0}(a \mid u_{\text{instruction}}, s) = \begin{cases} 0 & \text{if } a \notin s \\ \delta_{[\![u]\!](a)} & \text{if } [\![u]\!] \in s \\ \frac{1}{|s|} & \text{otherwise} \end{cases} \tag{8}$$

where $\delta_{[\![u]\!](a)}$ represents the meaning of u, evaluating to one when utterance u grounds to a and zero otherwise. An instruction is a *partial policy*: it designates an action to take in a subset of states.

4.3 Formalizing descriptions

Rather than mapping to a specific action, descriptions provide information about the reward function [24, 25, 46]. Following [88], we model descriptions as providing the reward of a single feature, similar to feature queries [31]. Descriptions are thus a tuple: a one-hot binary feature vector and a scalar value, $\langle \mathbb{1}_K, \mathbb{R} \rangle$. These are messages like ⟨Blue, -2⟩. In this work, we consider the set of 6 features $\times$ 5 values in $[-2, -1, 0, 1, 2]$, yielding 30 descriptive utterances. Formally, L_0 "rules out" inconsistent hypotheses about reward weights w:

$$L_0(w \mid u_{\text{description}}) \propto \delta_{[\![u]\!](w)} P(w) \tag{9}$$

where $\delta_{[\![u]\!](w)}$ represents the meaning of u, evaluating to one when u is true of w and zero otherwise. In this work, we assume $P(w)$ is uniform and there is no correlation between weights. The listener then marginalizes over possible reward functions to choose an action:

$$\pi_{L_0}(a \mid u_{\text{description}}, s) \propto \exp\{\beta_{L_0} \cdot \sum_w R(a, w) L_0(w \mid u))\} \tag{10}$$

where β_{L_0} is again an inverse temperature parameter.[4]

4.4 Comparing instructions and descriptions

Prior work suggests that humans use a mix of instructions and descriptions [24, 25]. What modulates this—when should a rational speaker prefer instructions over descriptions? To explore the effects of horizon on utterance utility, we simulate a nearly-optimal speaker ($\beta_{S_1} = 10$). Fig. 1A shows our bandit setting. We assume the listener begins with a uniform prior over reward weights throughout, and use states consisting of three unique actions (giving 84 possible states).

Fig 2A shows how the speaker's choice of utterance varies as the horizon lengthens. At short horizons, the speaker optimizes for present rewards (Eq. 5) and chooses instructions and descriptions that target the "Spotted Red" action. At longer horizons, future rewards (Eq. 6) play a larger role, and the speaker blends the two objectives (Eq. 7) by describing the highly negative blue feature. Finally, at sufficiently long horizons, future rewards dominate and it settles on describing the green feature—which is irrelevant to the start state, but the most important feature for generalization. To quantify how rational speakers should use instructions and descriptions, we repeat the task for all 84 start states using horizons ranging 1-10 and different utterance sets. Fig 2B plots rewards for speakers with access to *only* instructions *or* descriptions, illustrating why this shift from instructions to descriptions occurs. Instructions outperform descriptions at short horizons (achieving the theoretical maximum average reward of 1.75); as the horizon lengthens, however, descriptions generalize better.[5] Overall, as the horizon lengthens, speakers with access to both instructions and descriptions choose descriptions exclusively, producing highly generalizable information (see Appendix A).

4.5 Learning from utterances: a pragmatic listener

We now ask how the listener should *learn* from the speaker's utterance. Following RSA (Section 3), a pragmatic listener L_1 inverts the speaker model to infer their latent reward function. This approach is closely related to inverse reward design [4]: because the speaker maximizes rewards (Eq. 7), our pragmatic listener effectively treats utterances as *proxy rewards* [89] specified in language.

Known horizon. If the speaker's horizon H is known, we can write a L_1 listener as:

$$L_1(w \mid s, u, H) \propto S_1(u \mid w, s, H) P(w) \tag{11}$$

Given an instruction, L_1 infers the reward weights that would make such an instruction optimal [13, 14]; given a description, L_1 can recover information about features that were not mentioned [24, 25]. The L_1 listener then chooses actions by substituting this posterior belief into Eq. 10. In practice,

[4] Due to the recursive structure of speaker and listener, simultaneously varying β_{S_1} and β_{L_0} introduces identifiability issues. We therefore fix $\beta_{L_0} = 3$ throughout this work, and only vary β_{S_1}. This implies that all speakers address the same listener, but do so with different degrees of optimality.

[5] Because our states consist of only three actions, it is often possible to find a description that uniquely identifies the best action. This allows descriptions to perform nearly as well as instructions at $H = 1$. However, as the number of available actions increases, instructions become increasingly advantageous; see Appendix A.

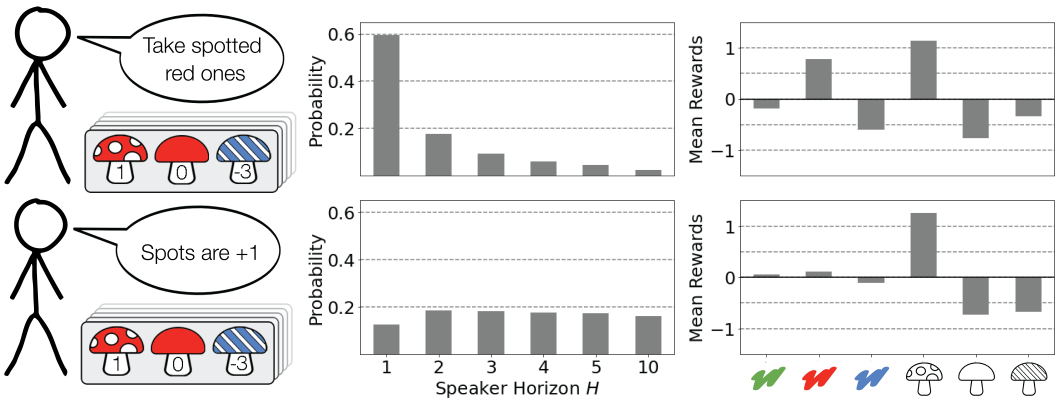

Figure 3: Posteriors from latent-horizon pragmatic inference (Eq. 12) for instructions and descriptions. Left: both utterances suggest the listener should take the spotted red mushroom in this state. Center: an instruction (top) suggests the speaker is short-horizon, while a description (bottom) suggests they are long-horizon. Right: this leads to different inference about the speaker's reward function (see Appendix A for full posteriors).

however, the horizon is not known, and so this approach is not feasible for real-world applications. This model instead serves as a theoretically-optimal baseline, and we next consider three practical ways of handling this uncertainty.

Assuming the horizon. Perhaps the most straightforward approach is to simply assume a speaker horizon. However, prior work has highlighted the risks of assuming a human is actively trying to teach [38], and suggests that the safest approach is to assume they are not. To test the effects of such mis-specification, in Section 5 we define two pragmatic listener models which assume a short ($H = 1$) or long ($H = 4$) horizon. A pragmatic listener assuming $H = 1$ will *constrain* inference: it will assume the utterance reflects only the speaker's preference between actions in the present state, and thus generalize conservatively. In contrast, a pragmatic listener assuming $H \gg 1$ expects the utterance to generalize broadly, risking overfitting.

Inferring the horizon. To mitigate the risk of horizon mis-specification, we can instead assume the speaker's horizon is unknown. Given an utterance, the *latent horizon* pragmatic listener jointly infers both their horizon and rewards, then marginalizes out the horizon:

$$L_1(w \mid s, u) \propto \sum_H S_1(u \mid w, s, H) P(H) P(w) \tag{12}$$

Intuitively, because short-horizon speakers prefer instructions (and long-horizon speakers prefer descriptions, Fig. 2A), the latent-horizon pragmatic listener can use the utterance type to infer the speaker's horizon and determine the appropriate scope of generalization. To demonstrate this, we simulate a pragmatic listener with a uniform prior over $H \in [1, 2, 3, 4, 5, 10]$. Fig. 3 shows example utterances and resulting inference about the speaker's rewards and latent horizon. Crucially, while both example utterances indicate a preference for the spotted red mushroom, the description suggests the speaker is $H > 1$ and uniquely identifies the spotted feature as high-value.

5 Behavioral experiment

To validate our theoretical models, we collected a behavioral dataset. Participants played the role of a mushroom foraging guide and produced utterances for tourists to help them choose good mushrooms. We manipulated the speaker's horizon by varying tourists' itineraries: each tourist was shown visiting a different visible mushroom patch, plus a variable number of unknown future patches (0, 1, or 3, matching horizons 1, 2, and 4 respectively). In the following sections, we analyze the participants' choice of utterances, compare the resulting inference from different listener models, and show how this socially-learned information can accelerate traditional reinforcement learning.[6]

[6]This study was approved by the Princeton IRB. The full experiment can be viewed at `https://pragmatic-bandits.herokuapp.com`.

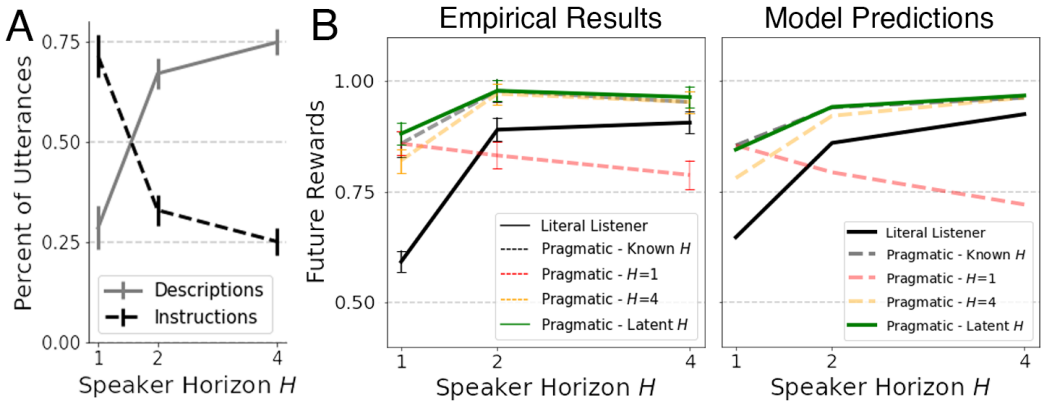

Figure 4: Results from behavioral experiment. **A**: Participants preferred instructions when $H = 1$ and descriptions otherwise. **B**, left: empirical future rewards (Eq. 6) from human utterances. As the horizon lengthens, humans choose more generalizable utterances; our latent-horizon listener successfully recovers their reward function. Right: simulations using our speaker model with $\beta_{S_1} = 3$. All error bars show 95% CI.

5.1 Experiment setup

We recruited 119 participants on the Prolific experimental platform (`prolific.co`). Participants were trained and tested on the game dynamics then advised a total of 28 tourists. They were told to consider the tourists' itinerary and choose "the most helpful utterance" from drop-down menus allowing them to specify an instruction or description. Because the space of possible descriptions (30) is larger than possible instructions (9), choosing a description required more effort than choosing an instruction. To equalize this, we reduced the set of descriptions by removing neutral features ("red" and "solid") and the 0 value, yielding $4 \times 4 = 16$ possible descriptive utterances. For the remainder of the paper, all pragmatic listeners assume the speaker chooses from this reduced utterance set. Participants received no feedback on how their utterances affected tourists' behaviors, ensuring they chose utterances according to their own sense of how to help. After screening out participants who failed comprehension or attention checks, we were left with 99 participants who produced a total of 2772 utterances. For more details on the data collection, see Appendix B.1.

We next use this set of utterances to explore value alignment (whether our models can infer the speakers' reward function) and then integrated social and reinforcement learning (whether this socially-acquired information can reduce regret in traditional RL).

5.2 Inferring rewards: value alignment from language

Overall, our results validated our theoretical models. First, participants were sensitive to the horizon manipulation: there was a statistically-significant shift in their utterance choices between horizons (for $H = 1$ to 2, $\chi^2(26, 1856) = 336.1, p < .001$; for $H = 2$ to 4, $\chi^2(26, 1833) = 40.3, p = .04$). Almost all participants (96 out of 99) used a mix of instructions and descriptions, favoring instructions at $H = 1$ and descriptions at $H > 1$ (Fig. 4A). This led to lower literal future rewards when $H = 1$, and higher future rewards at longer horizons (Fig 4B, "Empirical Results"). To calibrate our pragmatic listeners, we tested $\beta_{S_1} \in [1, 10]$ and found that $\beta_{S_1} = 3$ optimized Known H and Latent H listeners (see Appendix B.3 for details). Simulating utterances produced by our speaker model and resulting pragmatic inference shows a close match to theoretical predictions (Fig. 4B, "Model Predictions", see Appendix B.4 for details).

Our pragmatic listeners offered statistically-significant improvements over the literal listener (Table 1). These gains were particularly large when the speaker has a short horizon, matching our simulations. Somewhat surprisingly, our Latent H model ($M = 0.94$, $SD = 0.39$) outperformed all other models, including the Known H model ($M = .93$, $SD = .40$), by a significant albeit small margin (mean difference 0.01, paired-samples t-test $t(2771) = 4.18, p < .001$; see Appendix C for other pairwise tests). This is particularly notable because it underscores the inevitability of misspecification [38]: even when we experimentally controlled the horizon, participants did not perfectly follow our theory,

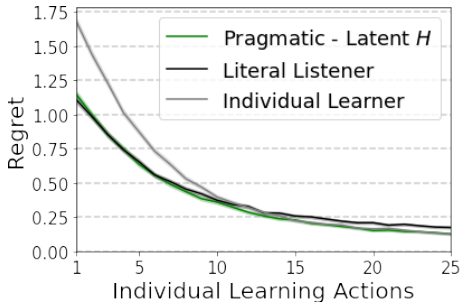

Figure 5: Regret with and without social information from our behavioral experiment.

Listener	Future rewards from utterance	Regret after 25 actions
Individual	-	12.14
Literal	.79	10.23
Prag - Known H	.93	**9.42***
Prag - $H = 1$	.83	9.76
Prag - $H = 4$	.91	9.67
Prag - Latent H	**.94*****	9.55

Table 1: Results from human utterances. Higher rewards (Section 5.2, Eq. 6) and lower regret (Section 5.3) are better. Bolded results are significantly better.

leading our Latent H model to outperform. We also confirmed the risks of *assuming* a horizon: the two fixed-horizon listeners ($H = 1$ and $H = 4$) underperformed the Latent H listener.

Despite these successes, we found a notable discrepancy between theoretical predictions and empirical results, suggesting a possible future refinement. As described in [88], utility-maximizing speakers agnostic to the truth of an utterance may send messages with exaggerated values (e.g. preferring the false utterance ⟨Spotted, +2⟩ to the true utterance ⟨Spotted, +1⟩). In our experiment, however, participants regularly chose the lower-reward, true utterance (Appendix B). Breaking out results by utterance type indicates that pragmatic gains come primarily on instructions (Appendix D). This suggests that our reward-design objective is in fact too weak: participants' tendency to use true descriptions licenses stronger inference, which we return to in the discussion.

5.3 Reducing regret: integrating social and reinforcement learning

We now explore how this form of social learning could augment traditional RL approaches. In particular, we take the inferences from our listeners in the previous section and use them as a prior for reinforcement learning (or in the case of literal instructions, we integrate them into the learner's policy; see Appendix E for details on the experimental setup). We then use Thompson sampling [90–92] to learn the reward function and compare regret over the course of learning for five independent learning trials on each context-utterance pair. We report results for our listener agents, as well as an agent with no social information (the "Individual" agent; Fig. 5, Table 1).

We find that social information can substantially reduce regret. Intriguingly, when comparing regret, our Known H pragmatic listener now achieves the best results; this is likely due to the greater uncertainty inherent in the Latent H inference. To confirm these differences are statistically significant, we use a linear regression to predict regret with a fixed effect of listener and random effects for each of the 2772 utterance-context pairs, again comparing the Latent H listener to all other listeners. This confirms that the Known H listener achieves lower regret than the Latent H listener ($\beta = -.12$, $t(80400) = -2.14, p = .03$), and all other models suffer higher regret (see Appendix C).

6 Discussion

We formalized the challenge of using language to teach an agent to act on our behalf [13, 14, 43, 44, 84] in a bandit setting, introducing the notion of a *horizon* to reflect the agent's degree of autonomy. We find *instructions* are optimal for low-autonomy settings, while *descriptions* are optimal for high autonomy. This distinction allows a pragmatic listener to jointly infer the speaker's horizon and reward function, reducing misspecification risk [38]. Our behavioral experiment supports our theory, demonstrating the benefit of learning from language for value alignment [1–3] and traditional RL [26].

We note several limitations and future directions. First, human participants preferred *truthful* descriptions; this licenses stronger pragmatics [93] such as combining our model with epistemic objectives [64, 65]. Second, we assumed language groundings were known, but future work could incorporate uncertainty [76, 77] or learn them [24, 25, 75]. Third, we studied a simple contextual bandit setting; sequential decision settings will require models of planning, and may consider additional language such as instructions at different levels of abstraction or *transition* descriptions [94]. Lastly,

we modeled a simple interaction consisting of a single utterance. More naturalistic interactions would allow multiple utterances, bidirectional dialogue, or interleaved action and communication [43, 44].

Our work can inform agent design by clarifying how and when different forms of linguistic input can be useful. Instruction following may be optimal when autonomy is unnecessary or preferences are non-Markovian [95]. However, if autonomy is desired [1–3, 13, 14, 38, 43, 44], agents should be equipped to understand descriptions of the world or our preferences [21–25]. Finally, pragmatic agents can *infer* whether preferences are local or general. This suggests that learning from a wide range of language is a promising approach for both value alignment and RL more broadly.

Acknowledgments and Disclosure of Funding

We thank Rachit Dubey, Karthik Narasimhan, and Carlos Correa for helpful discussions. TRS is supported by the NDSEG Fellowship Program and RDH is supported by the NSF (grant #1911835). This work was additionally supported by a John Templeton Foundation grant to TLG (#61454) and a grant from the Hirji Wigglesworth Family Foundation to DHM.

References

[1] Iason Gabriel. Artificial intelligence, values, and alignment. *Minds and Machines*, 30(3): 411–437, September 2020.

[2] Stuart Russell. *Human Compatible: Artificial Intelligence and the Problem of Control*. Viking, USA, 2019.

[3] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in AI safety. *arXiv preprint arXiv:1606.06565*, 2016.

[4] Dylan Hadfield-Menell, Smitha Milli, Pieter Abbeel, Stuart Russell, and Anca Dragan. Inverse reward design. In *Advances in Neural Information Processing Systems*, 2017.

[5] W Bradley Knox, Alessandro Allievi, Holger Banzhaf, Felix Schmitt, and Peter Stone. Reward (mis) design for autonomous driving. *arXiv preprint arXiv:2104.13906*, 2021.

[6] Terry Winograd. Understanding natural language. *Cognitive Psychology*, 3(1):1–191, 1972.

[7] Stefanie Tellex, Thomas Kollar, Steven Dickerson, Matthew R. Walter, Ashis Gopal Banerjee, Seth Teller, and Nicholas Roy. Understanding natural language commands for robotic navigation and mobile manipulation. In *AAAI Conference on Artificial Intelligence*, 2011.

[8] David Chen and Raymond Mooney. Learning to interpret natural language navigation instructions from observations. In *AAAI Conference on Artificial Intelligence*, 2011.

[9] James MacGlashan, Monica Babes-Vroman, Marie desJardins, Michael L Littman, Smaranda Muresan, Shawn Squire, Stefanie Tellex, Dilip Arumugam, and Lei Yang. Grounding English commands to reward functions. In *Robotics: Science and Systems*, 2015.

[10] Justin Fu, Anoop Korattikara, Sergey Levine, and Sergio Guadarrama. From language to goals: Inverse reinforcement learning for vision-based instruction following. In *International Conference on Learning Representations*, 2018.

[11] Dzmitry Bahdanau, Felix Hill, Jan Leike, Edward Hughes, Seyed Arian Hosseini, Pushmeet Kohli, and Edward Grefenstette. Learning to understand goal specifications by modelling reward. In *International Conference on Learning Representations*, 2019.

[12] Pratyusha Sharma, Balakumar Sundaralingam, Valts Blukis, Chris Paxton, Tucker Hermans, Antonio Torralba, Jacob Andreas, and Dieter Fox. Correcting robot plans with natural language feedback. *arXiv preprint arXiv:2204.05186*, 2022.

[13] Hong Jun Jeon, Smitha Milli, and Anca Dragan. Reward-rational (implicit) choice: A unifying formalism for reward learning. *Advances in Neural Information Processing Systems*, 2020.

[14] Smitha Milli, Dylan Hadfield-Menell, Anca Dragan, and Stuart Russell. Should robots be obedient? In *International Joint Conference on Artificial Intelligence*, 2017.

[15] Richard Maclin and Jude W. Shavlik. Incorporating advice into agents that learn from reinforcements. In *AAAI Conference on Artificial Intelligence*, 1994.

[16] Gregory Kuhlmann, Peter Stone, Raymond Mooney, and Jude Shavlik. Guiding a reinforcement learner with natural language advice: Initial results in RoboCup soccer. *AAAI Workshop - Technical Report*, 06 2004.

[17] Prasoon Goyal, Scott Niekum, and Raymond J Mooney. Using natural language for reward shaping in reinforcement learning. In *International Joint Conference on Artificial Intelligence*, 2019.

[18] J Luketina, N Nardelli, G Farquhar, J Foerster, J Andreas, E Grefenstette, S Whiteson, and T Rocktäschel. A survey of reinforcement learning informed by natural language. In *International Joint Conference on Artificial Intelligence*, 2019.

[19] Stefanie Tellex, Nakul Gopalan, Hadas Kress-Gazit, and Cynthia Matuszek. Robots that use language. *Annual Review of Control, Robotics, and Autonomous Systems*, 3(1):25–55, 2020.

[20] SRK Branavan, David Silver, and Regina Barzilay. Learning to win by reading manuals in a Monte-Carlo framework. *Journal of Artificial Intelligence Research*, 43:661–704, 2012.

[21] Karthik Narasimhan, Regina Barzilay, and Tommi Jaakkola. Grounding language for transfer in deep reinforcement learning. *Journal of Artificial Intelligence Research*, 63:849–874, 2018.

[22] Victor Zhong, Tim Rocktäschel, and Edward Grefenstette. RTFM: Generalising to novel environment dynamics via reading. *arXiv preprint arXiv:1910.08210*, 2019.

[23] Austin W Hanjie, Victor Y Zhong, and Karthik Narasimhan. Grounding language to entities and dynamics for generalization in reinforcement learning. In *International Conference on Machine Learning*, pages 4051–4062. PMLR, 2021.

[24] Theodore R Sumers, Mark K Ho, Robert D Hawkins, Karthik Narasimhan, and Thomas L Griffiths. Learning rewards from linguistic feedback. In *AAAI Conference on Artificial Intelligence*, 2021.

[25] Jessy Lin, Daniel Fried, Dan Klein, and Anca Dragan. Inferring rewards from language in context. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, 2022.

[26] Richard S Sutton and Andrew G Barto. *Reinforcement Learning: An Introduction*. MIT press, 2018.

[27] Andrew Y Ng and Stuart J Russell. Algorithms for inverse reinforcement learning. In *International Conference on Machine Learning*, 2000.

[28] Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *International Conference on Machine Learning*, 2004.

[29] Douglas B Markant and Todd M Gureckis. Is it better to select or to receive? Learning via active and passive hypothesis testing. *Journal of Experimental Psychology: General*, 143(1):94, 2014.

[30] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep Reinforcement Learning from Human Preferences. In *Advances in Neural Information Processing Systems*, 2017.

[31] Chandrayee Basu, Mukesh Singhal, and Anca D Dragan. Learning from richer human guidance: Augmenting comparison-based learning with feature queries. In *13th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2018.

[32] Natalia Vélez and Hyowon Gweon. Learning from other minds: An optimistic critique of reinforcement learning models of social learning. *Current Opinion in Behavioral Sciences*, 38: 110–115, 2021.

[33] Hyowon Gweon. Inferential social learning: Cognitive foundations of human social learning and teaching. *Trends in Cognitive Sciences*, 25(10):896–910, 2021.

[34] Pei Wang, Junqi Wang, Pushpi Paranamana, and Patrick Shafto. A mathematical theory of cooperative communication. *Advances in Neural Information Processing Systems*, 2020.

[35] Patrick Shafto, Noah D. Goodman, and Thomas L. Griffiths. A rational account of pedagogical reasoning: Teaching by, and learning from, examples. *Cognitive Psychology*, 71:55 – 89, 2014.

[36] Mark K Ho, Michael Littman, James MacGlashan, Fiery Cushman, and Joseph L Austerweil. Showing versus Doing: Teaching by Demonstration. In *Advances in Neural Information Processing Systems*, 2016.

[37] Mark K. Ho, Fiery Cushman, Michael L. Littman, and Joseph L. Austerweil. Communication in action: Planning and interpreting communicative demonstrations. *Journal of Experimental Psychology: General*, 150(11):2246–2272, November 2021.

[38] Smitha Milli and Anca D Dragan. Literal or pedagogic human? Analyzing human model misspecification in objective learning. In *Uncertainty in Artificial Intelligence*, 2020.

[39] W. Bradley Knox and Peter Stone. Interactively shaping agents via human reinforcement: The TAMER framework. In *Proceedings of the Fifth International Conference on Knowledge Capture*, page 9–16, New York, NY, USA, 2009. Association for Computing Machinery.

[40] James MacGlashan, Mark K Ho, Robert Loftin, Bei Peng, Guan Wang, David L. Roberts, Matthew E. Taylor, and Michael L. Littman. Interactive learning from policy-dependent human feedback. In *International Conference on Machine Learning*, 2017.

[41] Mark K. Ho, Fiery Cushman, Michael L. Littman, and Joseph L. Austerweil. People teach with rewards and punishments as communication, not reinforcements. *Journal of Experimental Psychology: General*, 148(3):520–549, March 2019.

[42] Brian D. Ziebart, Andrew Maas, J. Andrew Bagnell, and Anind K. Dey. Maximum entropy inverse reinforcement learning. In *AAAI Conference on Artificial Intelligence*, 2008.

[43] Dylan Hadfield-Menell, Stuart J Russell, Pieter Abbeel, and Anca Dragan. Cooperative inverse reinforcement learning. In *Advances in Neural Information Processing Systems 29*. 2016.

[44] Jaime F Fisac, Monica A Gates, Jessica B Hamrick, Chang Liu, Dylan Hadfield-Menell, Malayandi Palaniappan, Dhruv Malik, S Shankar Sastry, Thomas L Griffiths, and Anca D Dragan. Pragmatic-pedagogic value alignment. In *Robotics Research*, pages 49–57. Springer, 2020.

[45] Deepak Ramachandran and Eyal Amir. Bayesian inverse reinforcement learning. In *International Joint Conference on Artificial Intelligence*, 2007.

[46] Silvia Tulli, Francisco S. Melo, Ana Paiva, and Mohamed Chetouani. Learning from explanations with maximum likelihood inverse reinforcement learning. March 2022.

[47] Christian Daniel, Oliver Kroemer, Malte Viering, Jan Metz, and Jan Peters. Active reward learning with a novel acquisition function. *Auton. Robots*, 39(3):389–405, October 2015.

[48] Dorsa Sadigh, Anca D Dragan, Shankar Sastry, and Sanjit A Seshia. Active preference-based learning of reward functions. In *Robotics: Science and Systems*, 2017.

[49] Malayandi Palan, Gleb Shevchuk, Nicholas Charles Landolfi, and Dorsa Sadigh. Learning reward functions by integrating human demonstrations and preferences. In *Robotics: Science and Systems*, 2019.

[50] Valerie J. Shute. Focus on formative feedback. *Review of Educational Research*, 78(1):153–189, 2008.

[51] A.A. Lipnevich and J.K. Smith. Effects of differential feedback on students' examination performance. *Journal of Experimental Psychology: Applied*, 15(4):319–333, 2009.

[52] John D Co-Reyes, Abhishek Gupta, Suvansh Sanjeev, Nick Altieri, John DeNero, Pieter Abbeel, and Sergey Levine. Meta-learning language-guided policy learning. In *International Conference on Learning Representations*, 2019.

[53] Pratyusha Sharma, Antonio Torralba, and Jacob Andreas. Skill induction and planning with latent language. *arXiv preprint arXiv:2110.01517*, 2021.

[54] Catherine Wong, Kevin M Ellis, Joshua Tenenbaum, and Jacob Andreas. Leveraging language to learn program abstractions and search heuristics. In *International Conference on Machine Learning*, 2021.

[55] Jacob Andreas, Dan Klein, and Sergey Levine. Learning with latent language. In *NAACL*, pages 2166–2179, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[56] Valerie Chen, Abhinav Gupta, and Kenneth Marino. Ask your humans: Using human instructions to improve generalization in reinforcement learning. In *International Conference on Learning Representations*, 2021.

[57] Khanh X Nguyen, Dipendra Misra, Robert Schapire, Miroslav Dudík, and Patrick Shafto. Interactive learning from activity description. In *International Conference on Machine Learning*, 2021.

[58] Yiding Jiang, Shixiang Shane Gu, Kevin P Murphy, and Chelsea Finn. Language as an abstraction for hierarchical deep reinforcement learning. *Advances in Neural Information Processing Systems*, 2019.

[59] Allen Z Ren, Bharat Govil, Tsung-Yen Yang, Karthik Narasimhan, and Anirudha Majumdar. Leveraging language for accelerated learning of tool manipulation. *arXiv preprint arXiv:2206.13074*, 2022.

[60] Cédric Colas, Tristan Karch, Nicolas Lair, Jean-Michel Dussoux, Clément Moulin-Frier, Peter Dominey, and Pierre-Yves Oudeyer. Language as a cognitive tool to imagine goals in curiosity driven exploration. *Advances in Neural Information Processing Systems*, 2020.

[61] Geoffrey Cideron, Mathieu Seurin, Florian Strub, and Olivier Pietquin. Higher: Improving instruction following with hindsight generation for experience replay. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 225–232. IEEE, 2020.

[62] Jesse Mu, Victor Zhong, Roberta Raileanu, Minqi Jiang, Noah Goodman, Tim Rocktäschel, and Edward Grefenstette. Improving intrinsic exploration with language abstractions. *arXiv preprint arXiv:2202.08938*, 2022.

[63] Allison C Tam, Neil C Rabinowitz, Andrew K Lampinen, Nicholas A Roy, Stephanie CY Chan, DJ Strouse, Jane X Wang, Andrea Banino, and Felix Hill. Semantic exploration from language abstractions and pretrained representations. *arXiv preprint arXiv:2204.05080*, 2022.

[64] Michael C Frank and Noah D Goodman. Predicting pragmatic reasoning in language games. *Science*, 336(6084):998–998, 2012.

[65] Noah D. Goodman and Michael C. Frank. Pragmatic language interpretation as probabilistic inference. *Trends in Cognitive Sciences*, 20(11):818 – 829, 2016.

[66] H. P. Grice. Logic and conversation. In *Syntax and Semantics: Vol. 3: Speech Acts*, pages 41–58. Academic Press, New York, 1975.

[67] Will Monroe, Robert XD Hawkins, Noah D Goodman, and Christopher Potts. Colors in context: A pragmatic neural model for grounded language understanding. *Transactions of the Association for Computational Linguistics*, 5:325–338, 2017.

[68] Jacob Andreas and Dan Klein. Reasoning about pragmatics with neural listeners and speakers. In *Empirical Methods in Natural Language Processing*, 2016.

[69] Allen Nie, Reuben Cohn-Gordon, and Christopher Potts. Pragmatic issue-sensitive image captioning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1924–1938, 2020.

[70] Daniel Fried, Jacob Andreas, and Dan Klein. Unified pragmatic models for generating and following instructions. In *NAACL*, June 2018.

[71] Erica J Yoon, Michael Henry Tessler, Noah D Goodman, and Michael C Frank. Polite speech emerges from competing social goals. *Open Mind*, 4:71–87, 2020.

[72] Justine T. Kao, Jean Y. Wu, Leon Bergen, and Noah D. Goodman. Nonliteral understanding of number words. *Proceedings of the National Academy of Sciences*, 111(33):12002–12007, 2014.

[73] Justine Kao, Leon Bergen, and Noah Goodman. Formalizing the pragmatics of metaphor understanding. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 36, 2014.

[74] Stevan Harnad. The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42(1-3): 335–346, 1990.

[75] Raymond J Mooney. Learning to connect language and perception. In *AAAI Conference on Artificial Intelligence*, 2008.

[76] Judith Degen, Robert D Hawkins, Caroline Graf, Elisa Kreiss, and Noah D Goodman. When redundancy is useful: A Bayesian approach to "overinformative" referring expressions. *Psychological Review*, 127(4):591, 2020.

[77] Robert D Hawkins, Michael Franke, Michael C Frank, Adele E Goldberg, Kenny Smith, Thomas L Griffiths, and Noah D Goodman. From partners to populations: A hierarchical Bayesian account of coordination and convention. *Psychological Review*, 2022.

[78] Thomas M Cover. *Elements of Information Theory*. John Wiley & Sons, 1999.

[79] Stephen C Levinson. *Pragmatics*. Cambridge university press, 1983.

[80] Stephen C Levinson. *Presumptive meanings: The theory of generalized conversational implicature*. MIT press, 2000.

[81] Noah D Goodman and Andreas Stuhlmüller. Knowledge and implicature: Modeling language understanding as social cognition. *Topics in Cognitive Science*, 5(1):173–184, 2013.

[82] Martin L Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., 1994.

[83] Tor Lattimore and Csaba Szepesvari. *Bandit Algorithms*. Cambridge University Press, 2020.

[84] Kareem Amin, Nan Jiang, and Satinder Singh. Repeated inverse reinforcement learning. *Advances in Neural Information Processing Systems*, 2017.

[85] Carlos Riquelme, George Tucker, and Jasper Snoek. Deep Bayesian bandits showdown: An empirical comparison of Bayesian deep networks for Thompson sampling. In *International Conference on Learning Representations*, 2018.

[86] Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandits with linear payoff functions. In *International Conference on Artificial Intelligence and Statistics*, volume 15, 2011.

[87] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, volume 24, 2011.

[88] Theodore Sumers, Robert Hawkins, Mark Ho, and Thomas Griffiths. Extending rational models of communication from beliefs to actions. In *Proceedings of the Annual Conference of the Cognitive Science Society*, 2021.

[89] Satinder Singh, Richard L Lewis, and Andrew G Barto. Where do rewards come from? In *Proceedings of the 31st Annual Conference of the Cognitive Science Society*, 2009.

[90] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3-4):285–294, 1933.

[91] Daniel Russo and Benjamin Van Roy. Learning to optimize via posterior sampling. *Mathematics of Operations Research*, 39(4):1221–1243, 2014.

[92] Daniel J Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, and Zheng Wen. A tutorial on Thompson sampling. *Foundations and Trends in Machine Learning*, 11(1):1–96, 2018.

[93] Rohin Shah, Noah Gundotra, Pieter Abbeel, and Anca Dragan. On the feasibility of learning, rather than assuming, human biases for reward inference. In *International Conference on Machine Learning*, volume 97, 2019.

[94] Anna N. Rafferty, Emma Brunskill, T. Griffiths, and Patrick Shafto. Faster teaching by POMDP planning. In *AIED*, 2011.

[95] David Abel, Will Dabney, Anna Harutyunyan, Mark K Ho, Michael Littman, Doina Precup, and Satinder Singh. On the expressivity of Markov reward. *Advances in Neural Information Processing Systems*, 34:7799–7812, 2021.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See the discussion (Section 6).
   (c) Did you discuss any potential negative societal impacts of your work? [No] Our work primarily theoretical and intended to have positive societal impact. We do not foresee any material risks at this time.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] See in particular details on our behavioral experiment consent / IRB (Appendix B)).
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A] Our work does not contain theoretical results.

(b) Did you include complete proofs of all theoretical results? [N/A] Our work does not contain theoretical results.

3. If you ran experiments...

(a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We will submit code in the supplemental material, and release on GitHub for the camera-ready.

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] We do not train any models, but we specify the parameter-setting procedures used for the human data in Appendix B.

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] All plots contain error bars where relevant and we conduct statistical testing to quantify uncertainty where appropriate.

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] All analyses were run on consumer hardware (e.g. a Macbook). We note this in Appendix E.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [N/A] We do not use any existing assets.

(b) Did you mention the license of the assets? [N/A] We do not use any existing assets.

(c) Did you include any new assets either in the supplemental material or as a URL? [Yes] We include the (anonymized) data from our experiment in the supplemental materials and will release via GitHub.

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] The consent form used is linked as part of the experiment.

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] We hash participant identifiers and include only information collected during the coruse of our experiment. This is discussed in Appendix B.

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [Yes] We link to the full (anonymized) experiment in Appendix B.

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A] We did not anticipate any participant risks; this is discussed in the consent form on the experiment.

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [Yes] See Appendix B.

A Speaker simulations and pragmatic inference

A.1 Instructions vs Descriptions

In the main text, we used a fixed number of available actions (a context S with $|S| = 3$ objects). Here, we further explore the effect of horizon on the choice of instructions vs. descriptions under different numbers of available actions. As in the main text, we assume that the speaker uses a near-optimal softmax temperature for clarity by setting $\beta_{S_1} = 10$.

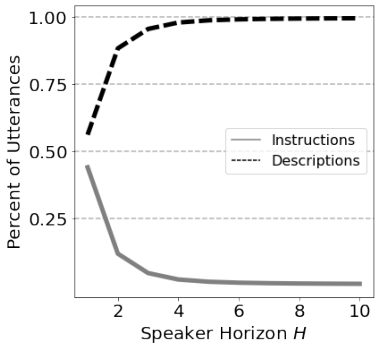

Figure S1: As noted in Section 4.4, speakers exhibit a strong preference for descriptions as their horizon lengthens.

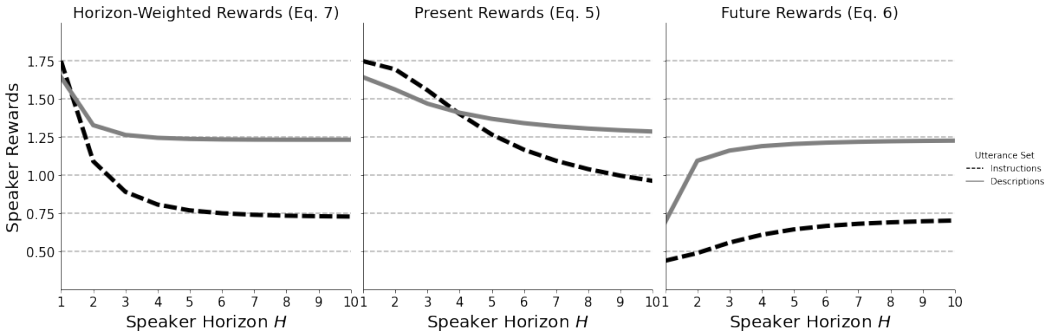

Figure S2: Breaking out speaker rewards (Fig. 2B) by reward type and utterance. "Horizon-Weighted Rewards" (left) is the same as Fig. 2B. Instructions afford high "Present Rewards" (center) but generalize poorly (low "Future Rewards", right). As a result, rational speakers with access to instructions only remain biased towards the present context even as their horizon lengthens. This can be seen by comparing "Present" and "Future" rewards at long horizons (e.g. $H = 10$). Description-only speakers exhibit little bias towards the present context ("Present" and "Future" rewards are nearly equal), while instruction-only speakers remain biased towards the present context ("Present" > "Future" rewards).

We find that instructions become more useful as the number of available actions increases. They can always uniquely select the best action in a given state (even when all nine possible objects are present), whereas it is not always possible to use a description to identify the best action.

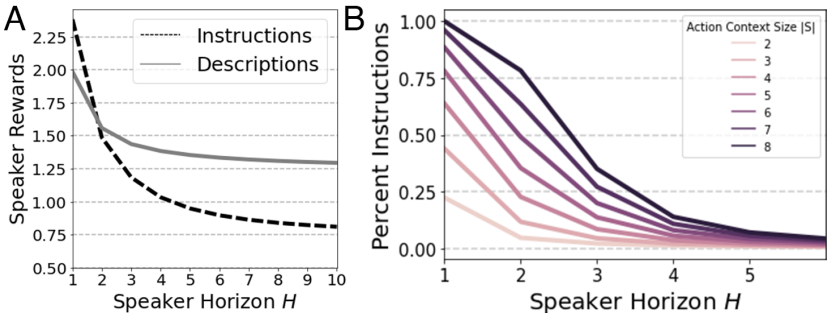

Figure S3: **A**: Same as Fig. 2B, but with a state-size $|S|$ of 5 instead of 3. At short horizons, the relative utility of instructions increases with state size (e.g. as the action space grows, instructions are more useful). **B**: Speaker's probability of using an instruction as a function of number of available actions $|S|$ and horizon H (note that Fig. S1 shows the curve for $|S| = 3$). As the number of actions increases, speakers prefer instructions.

Next, building on Fig. 3 in the main text, we include full posteriors over reward functions and additional utterances.

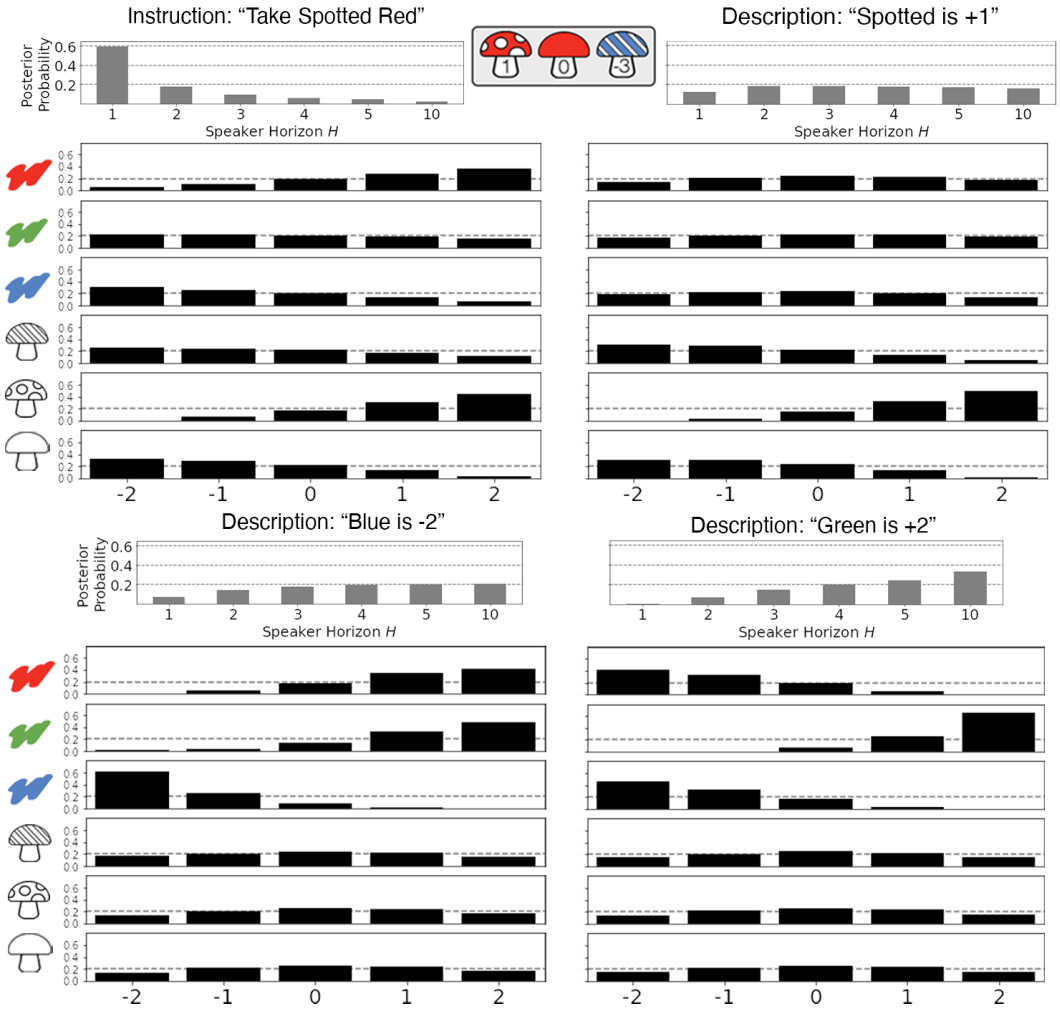

Figure S4: Same as Fig. 3, but with full utterance posteriors over possible reward functions and additional utterances. For utterance posteriors, the gray dashed line indicates the prior (e.g. uniform over all possible values). Note that descriptions suggest that unmentioned features are lower-magnitude (e.g. for the bottom-right “Green is +2” utterance, the listener infers that all textures—Striped, Spotted, and Solid—are unlikely to be -2 or +2). Finally, note that with all descriptive utterances, the listener assigns non-negligible probability mass to values *other* than the specified one (e.g. in the top-right, the listener infers a substantial probability that the Spotted feature is actually +2). This suggests that integrating a “truthfulness bias” could improve our models (see Section 5.2 for a discussion).

B Behavioral Experiment

The full experiment can be viewed at `pragmatic-bandits.herokuapp.com`. Note that the app is running on free dynos so you may need to wait 5-10 seconds for it to load.

B.1 Experiment Details

Participant compensation. Participants earned an average hourly wage of $12.10 and the total amount spent on participant compensation was $444. The mean time spent in the experiment was 18.5 minutes.

IRB approvals. This study was approved by the Princeton University IRB. All participants gave informed consent; the consent form can be seen at the experiment URL above. As described in the Checklist, no significant participant risks were anticipated.

Anonymized data. Anonymized data, including participant responses and free-form exit survey responses, is available in the supplementary zip file and will be released along with the code for this paper. Note that the worker IDs provided have been hashed to prevent re-identification of participants on the platform.

Trials. We split our 84 states into 3 sets of 28; each participant saw one of these sets. Each participant additionally saw 8 "attention check" trials (constant across all participants). These "attention checks" forced the participant to use a description with a pre-selected feature (4 "Spotted" and 4 "Striped"). The participant then chose a value from $[-1, +1]$. The trials were selected to ensure that the true value would lead the learner to choose a good mushroom. Participants who failed to select the true value on at least 7/8 trials (e.g. >75% of the time) were still paid the full bonus, but their data was excluded from the analysis. We further exclude the attention check trials from the analysis.

Feature Randomization. To avoid saliency biases (e.g. color may be more salient than texture), mushroom feature values were randomized across participants. Fig. S5 shows one example of an alternative featurization scheme. Note that all responses were converted back the the "canonical" feature map shown in Fig. 1 for analysis.

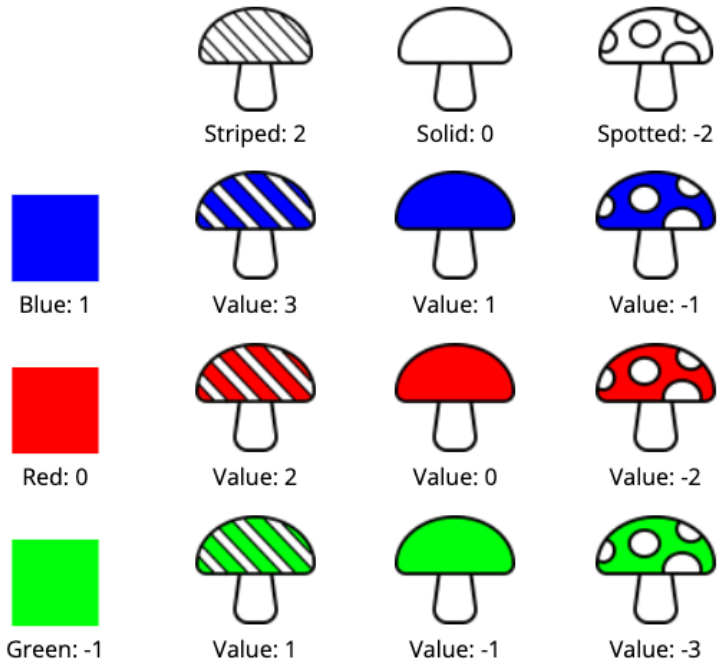

Figure S5: Throughout the experiment, participants could trigger a pop-up giving them the value of all features (and all actions). Note that features were randomized to avoid saliency biases, and so this set of feature values does not match Fig. 1.

Instructions We include several screenshots of key instruction pages, but recommend viewing the full experiment at `pragmatic-bandits.herokuapp.com` for details.

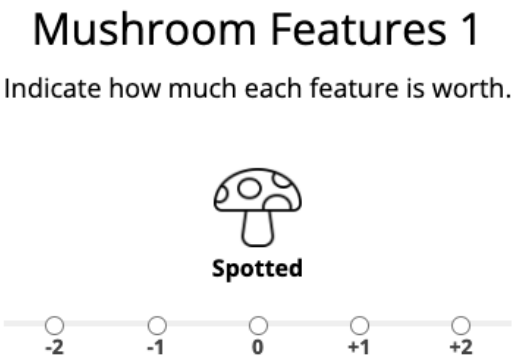

Figure S6: One example quiz question. To ensure comprehension of the contextual bandit setup, participants were tested on their knowledge of all features. If they failed the quiz, the experiment terminated early and they earned $2; if they passed, they completed the full experiment and earned $4.

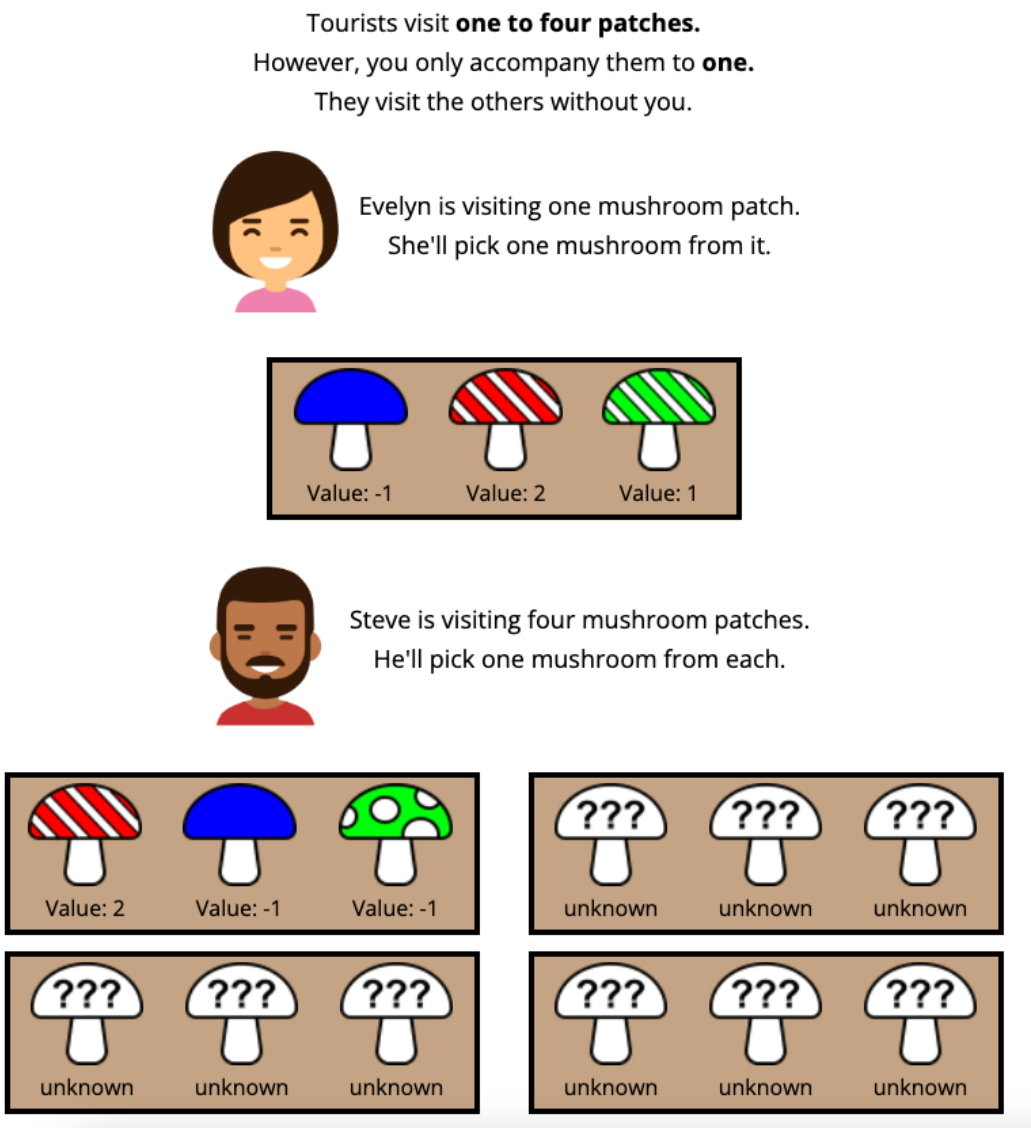

Figure S7: Experiment instructions: introducing the notion of the *horizon*. Participants were told they could only accompany the tourist to one patch, but that depending on their itinerary, tourists could go on to visit other (unknown) patches afterwards.

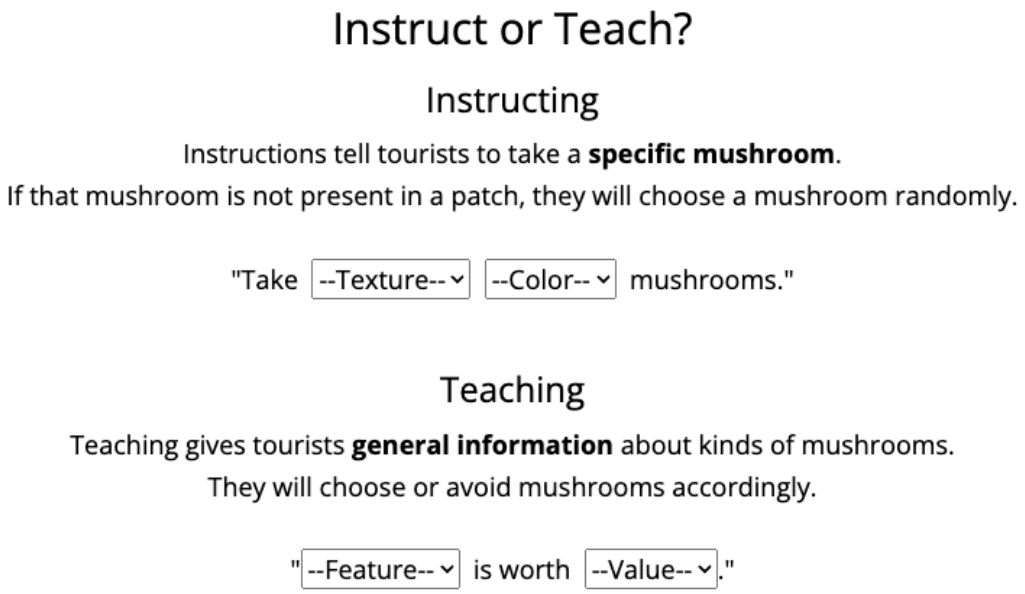

Figure S8: Experiment instructions: introducing the notion of instructions and descriptions. Participants were told to consider the tourist's itinerary (Fig. S7) and help the tourist pick good mushrooms throughout their visit.

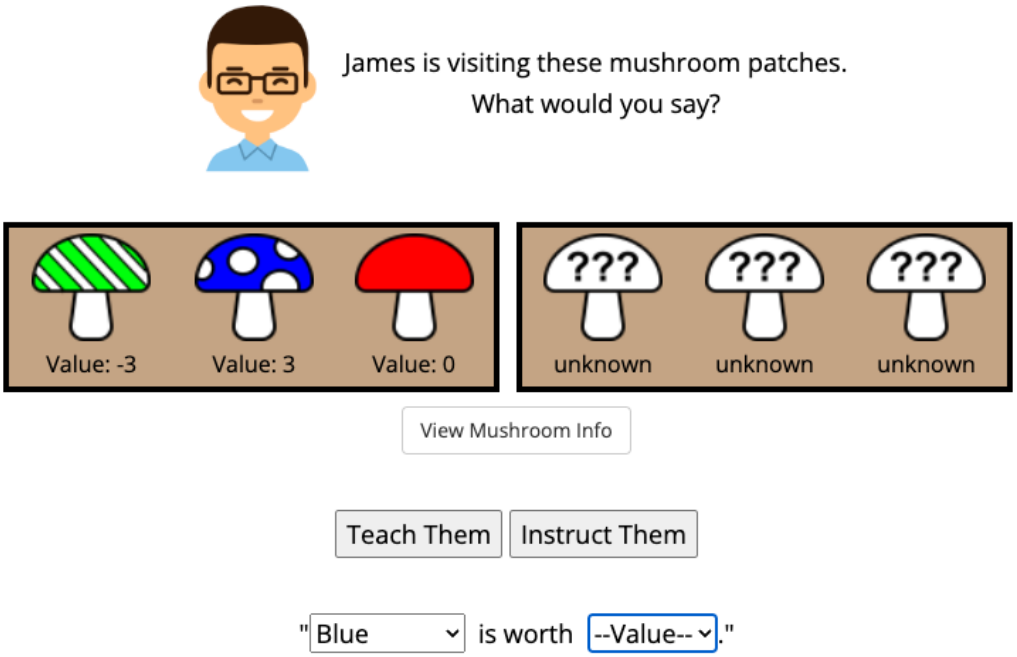

Figure S9: One example trial from the experiment. Clicking the "Teach Them" button revealed drop-down menus to select a "Description" utterance, while clicking "Instruct Them" yielded menus to select an "Instruction."

B.2 Participant Utterance Choices

The supplementary materials contain all participant responses (see the Experiment Analysis Jupyter notebook for analysis code). Here, we summarize some of the key patterns in the data.

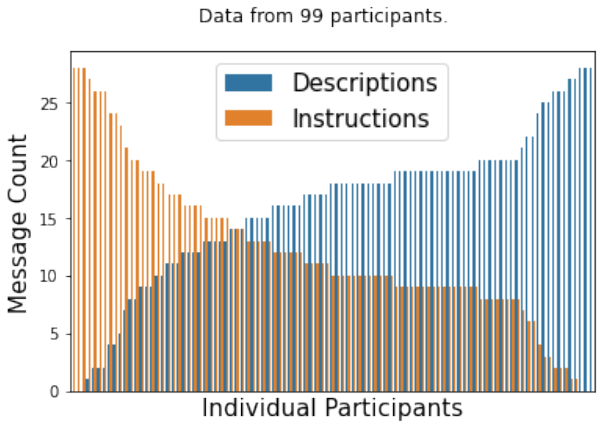

Figure S10: Breakdown of utterance types (instruction vs descriptions) for all participants. Most participants used a mix of both: 3 used only instructions, 3 used only descriptions, and 93 used at least one message of each type.

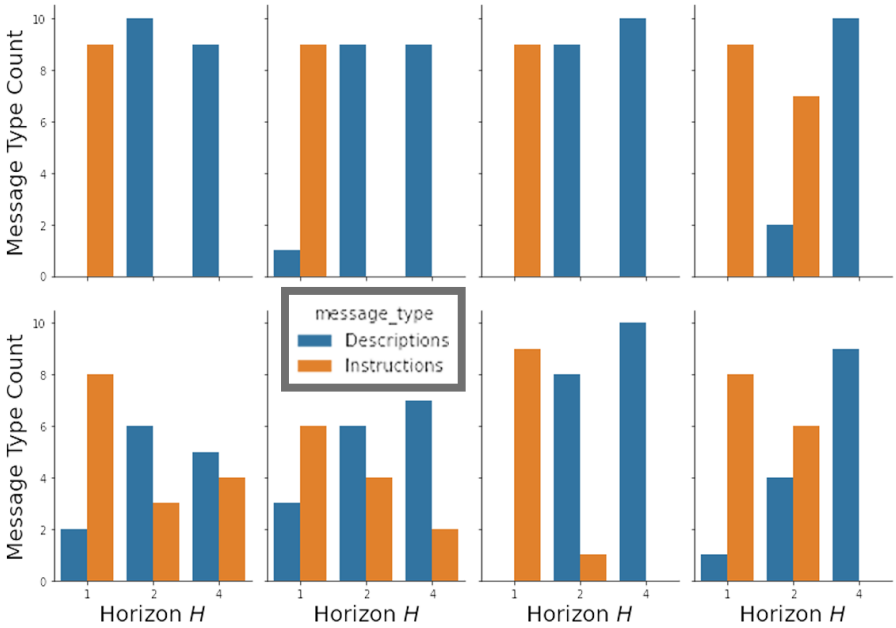

Figure S11: Instruction / description breakdown by horizon for 8 random participants. While individual preferences varied substantially, virtually all participants displayed an increasing preference for descriptions as the horizon increased.

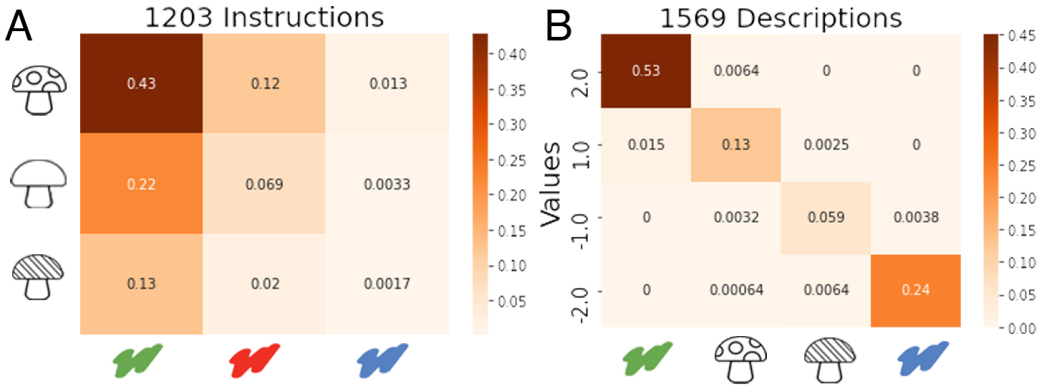

Figure S12: Within-type distribution of utterances chosen by participants. **A**: When giving instructions participants (unsurprisingly) almost always chose positive-reward actions, e.g. "Take the spotted green mushroom," in the top-left quadrant. **B**: When giving descriptions, participants almost always chose *true* utterances (e.g. "Spotted is +1"), even though our reward-maximizing model predicts exaggeration (e.g. "Spotted is +2"). See Section 5.2 for discussion.

B.3 Choosing β_{S_1}

To choose a β_{S_1} for our behavioral experiment, we used a grid search over integers $\beta_{S_1} \in [1, 10]$ and evaluated our primary models (Pragmatic - Known H and Pragmatic - Latent H). We chose $\beta_{S_1} = 3$, which optimized future reward from the human data for both speakers. Note that while "Pragmatic - Known H" was numerically optimal at $\beta_{S_1} = 2$ (expected reward $= 0.9308, SD = 0.39$), there was not find a significant difference between this and $\beta_{S_1} = 3$ (expected reward $= 0.9295, SD = 0.40$); paired-samples $t(2771) = -1.70, p = 0.09$.

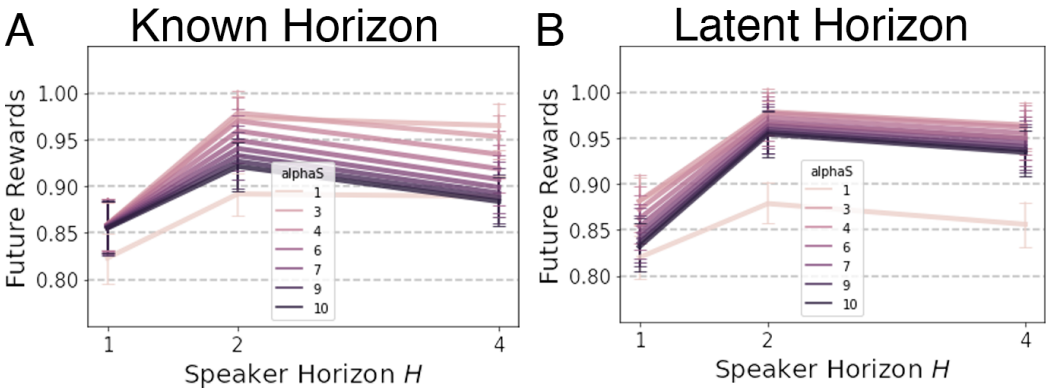

Figure S13: Performance of different pragmatic listener models as a function of horizon H and speaker-optimality β_{S_1}. Qualitatively, the Latent-H model (right) was less sensitive than the Known-H model (left).

β_{S_1}	Known H	Latent H
1	0.87	0.85
2	0.93	0.93
3	**0.93**	**0.94**
4	0.92	0.94
5	0.91	0.93
6	0.90	0.92
7	0.90	0.92
8	0.89	0.91
9	0.89	0.91
10	0.89	0.91

Table S1: Mean "Future Rewards" for our two primary models of interest as a function of the β_{S_1} parameter. Note that while "Pragmatic - Known H was slightly higher at $\beta_{S_1} = 2$ there was not find a significant difference between this value and $\beta_{S_1} = 3$, hence we use the latter to be consistent across both models.

B.4 Simulating model behavior

To compare the empirical pattern of utterances observed from humans in our experiment against the predictions of our theoretical speaker model, we use simulations to generate a distribution over utterances and directly compare the results (Fig. 4, "Model Predictions"). We set $\beta_{S_1} = 3$ as described above. The utterance set is composed of the 9 instructions and 16 descriptions defined in Section 5.1, for a total of 25 possible utterances.

First, for each $H \in [1, 2, 4]$ and each of the 84 states $s \in S$, we run the speaker model to produce a distribution over the 25 possible utterances (Eq. 7). We then calculate the literal future rewards resulting from each utterance (Eq. 8 for instructions and Eqs.9, 10 for descriptions). We then calculate the *expected future reward* obtained by the literal listener by weighting the rewards for each utterance by the probability of the speaker producing that utterance, and averaging over all 84 start states.

Similarly, to evaluate the pragmatic listener, we perform pragmatic inference over each utterance (Eq. 11) to recover the speaker's reward function, evaluate the future rewards (Eq. 6) from the resulting beliefs, and again weight by the speaker's distribution over utterances.

C Statistical testing

See the R notebook for statistical testing code.

C.1 Paired T-Tests (§ 5.2)

Comparison	Mean Difference	95% CI		t	df	p-val
vs. Pragmatic (Known H)	0.011	0.006	0.017	4.1798	2771	<.001
vs. Pragmatic ($H = 4$)	0.026	0.022	0.031	10.81	2771	<.001
vs. Pragmatic ($H = 1$)	0.11	0.10	0.13	20.545	2771	<.001
vs. Literal	0.15	0.13	0.16	23.364	2771	<.001

Table S2: Pairwise t-tests comparing the "Future Rewards" obtained by the Latent-H listener to other models for the 2772 utterances from our behavioral experiment. These results indicate that the Latent H model outperforms all other models.

Comparison	Mean Difference	95% CI		t	df	p-val
vs. Pragmatic (Known H)	-0.13	-0.15	-0.12	-19.854	2771	<.001
vs. Pragmatic ($H = 4$)	-0.12	-0.13	-0.11	-23.324	2771	<.001
vs. Pragmatic ($H = 1$)	-0.03	-0.05	-0.01	-3.202	2771	<.01

Table S3: Pairwise t-tests comparing the "Future Rewards" obtained by the Literal listener to the remaining other models for the 2772 utterances from our behavioral experiment. These results indicate that all pragmatic models outperform the Literal model.

C.2 Mixed-effects regression model (§ 5.3)

The following analysis tests for a significant difference in regret when using the model's social-learning posterior as a prior for individual learning (see Section E for details). Note that lower regret is better, so negative coefficients indicate better performance.

We dummy-coded our different models as a categorical variable with the Latent H listener as the reference level. We included random intercepts for each unique utterance from our experiment (e.g. for each of the 2772 utterances chosen by participants) to account for some utterances being systematically easier or harder than others. The resulting coefficients indicate that the Latent H listener outperformed all models except for the Known H model, which achieved slightly lower regret.

	Effect	Term	Estimate	Std Error	Statistic	df	p value
1	fixed	(Intercept)	9.55	0.05	195.66	14282.57	< 0.001
2	Fixed	Individual	2.60	0.06	45.41	80383.00	< 0.001
3	Fixed	Literal	0.68	0.06	11.93	80383.00	< 0.001
4	Fixed	Prag (Known H)	-0.12	0.06	-2.14	80383.00	0.03
5	Fixed	Prag ($H = 1$)	0.22	0.06	3.77	80383.00	< 0.001
6	Fixed	Prag ($H = 4$)	0.13	0.06	2.24	80383.00	0.03
7	Random Effect	sd__(Intercept)	1.44				
8	Residual	sd__Observation	4.76				

D When does pragmatic reasoning help?

In this section, we examine the utterances produced in the human experiment (Section 5 and Appendix B) to explore when, exactly, pragmatic reasoning is most useful. We analyze the performance of the Latent H pragmatic model in comparison to the Literal listener. Concretely, we take the 2772 utterances produced in our behavioral experiment and evaluate the "future rewards" (Eq. 6, the expected rewards over all possible states,) resulting from a literal interpretation of the utterance against those resulting from a pragmatic interpretation.

Utterance Type	Count	Mean Pragmatic Gain
Instruction	1203	$.42 \pm .23$
Description	1569	$-.07 \pm .22$

Table S4: Average pragmatic gain for different utterance types (+/- standard deviations). Pragmatics on instructions helps substantially by converting from partial policies to rewards, but pragmatics on descriptions marginally *reduces* the average reward obtained.

We find that under these conditions, pragmatic inference primarily helps with *instructions*, rather than descriptions (Table S4): converting a partial policy into inference over the reward function allows much stronger generalization. Across the 1203 instruction utterances in our experiment, the pragmatic listener achieved a large and statistically-significant gain ($M = .423, SD = .232$), $t(1202) = 63.34, p < .001$. In contrast, on the 1569 descriptive utterances, the pragmatic listener suffered a small but statistically-significant loss ($M = -.067, SD = .215$), $t(1568) = -12.29, p < .001$.

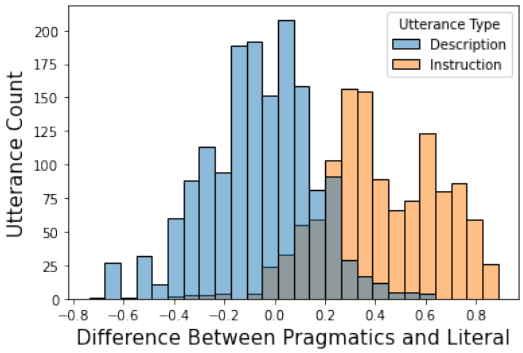

Figure S14: Distribution of pragmatic gain (Pragmatic listener with Latent H vs. Literal listener) for the 2772 utterances in our behavioral experiment. Pragmatic inference substantially improves rewards for instructions, but marginally reduces rewards for descriptions on expectation.

Analysis of utterance posteriors (Fig. S4) shows one notable disconnect with empirical human behavior regarding descriptive utterances. The pragmatic listener *does not* preserve the literal truth conditions of descriptive utterances: for the three descriptions shown in Fig. S4, the listener places substantial probability mass on values *other* than the specified one (e.g. believing that "Spotted is +1" suggests "Spotted is +2" is plausible). Yet in our experiment, participants almost always choose true utterances (see Fig. S12B). This suggests future work integrating truthfulness and reward objectives, effectively combining our current objective with classic Gricean notions.

E Social vs. individual reinforcement learning (§ 5.3)

To study the potential benefits of integrating social and reinforcement learning, we integrated the reward information learned from our behavioral experiment into a classic Thompson sampling individual learner in § 5.3. Here, we provide details on this integration. Code for these simulations can be found in the Supplemental Materials.

E.1 Individual learning: Thompson sampling in contextual bandits

We first define a simple individual learner in our linear contextual bandit setting using Thompson sampling [90–92]. The agent begins with a prior distribution over possible reward functions. At each timestep, they (1) observe a new state s_t consisting of three possible actions; (2) sample a reward weight vector w_t from their belief distribution over reward weights, and (3) act optimally according to that reward vector. They observe the reward of that action, and use this observation to update their beliefs for the next timestep.

We implement this algorithm using a Gaussian prior and likelihood function, assuming observation noise from a unit Gaussian. Thus, after taking action a, the agent receives rewards according to:

$$R(a) \sim \mathcal{N}(\phi(a)^\top w, 1) \tag{13}$$

We use a wide multivariate Gaussian prior: $w^0 \sim \mathcal{N}(0, \Sigma_0)$ where $\Sigma_0 = 5I$. After each action, we perform conjugate Bayesian updates to obtain a posterior (i.e. use Bayesian linear regression), which we use for for the next timestep.

We perform rejection sampling to ensure the sampled belief is compatible with the (discrete, bounded) reward function. We first sample a (continuous) weight vector from our multivariate Gaussian beliefs, then round the weights to integer values and reject the sample if any of the resulting values fall outside the range of possible reward values, $[-2, 2]$.

We note that these simulation parameters are arbitrary. Our aim is to demonstrate the general utility of social information to reduce regret *even when individual learning is entirely possible*. We thus defined a relatively straightforward, low-noise individual learning setting. However, we could easily make individual learning arbitrarily more difficult (e.g. by increasing the observation noise), which would in turn increase the relative value of social information.

E.2 Integrating pragmatic inference: Importance sampling

In order to integrate social information about the reward function, we incorporate an additional importance sampling step. Given a particular pragmatic model and a utterance-context-horizon tuple, we first use the pragmatic model to generate a *social* posterior over reward functions (Eq. 11 or 12). This defines a probability for every possible reward function (e.g. every reward weight vector w). We then initialize our individual learner as described above.

When the individual learner performs Thompson sampling, it now performs an additional importance sampling step. Rather than sample a *single* reward vector from its Gaussian prior, it samples a minimum of 100 possible reward vectors. As described above, it first discretizes these vectors then re-weights them according to the probability of each vector from its pragmatic social inference. Finally, it samples a single reward vector from this re-weighted set and uses this to choose an action.

E.3 Integrating literal information

We use a similar procedure to test individual learning with our literal listener. For descriptive utterances, we use the listener's posterior over reward functions (Eq. 9). However, because there are a handful of false utterances in the experimental data (e.g. "Spotted is -1"), using a hard constraint breaks the importance-sampling procedure described above. We therefore instead use soft-conditioning by setting a very low likelihood on inconsistent worlds ($\epsilon = 1^{-10}$) instead of ruling them out entirely. We use this posterior for importance sampling as described above.

For instructions, we modify the action selection step. We set the listener's policy to take the instructed action if available. If the action is not available, then they follow the Thompson sampling procedure described above. This is the simplest and most "obedient" interpretation of instructions [14]. We find that it yields rapid learning early on, as the instruction guides exploration. However, 57% of instructions designate sub-optimal actions (see Fig. S12A). A literal listener instructed to take one of these (e.g. "Take solid green mushrooms") is forced to continue taking them *even after* inferring spotted green mushrooms are likely worth more. This constraint on their policy eventually leads their regret to asymptote below the more flexible pragmatic learner (Fig. 5). As discussed in the main text and noted in prior work [14], more flexible approaches to instruction-following could avoid this pitfall.

E.4 Simulation details

All simulations were run on consumer hardware (a MacBook Pro). For each of the 2772 utterances in our behavioral experiment, we ran 5 independent Thompson sampling simulations, each spanning 25 timesteps. We repeated this process for each of the pragmatic listener models (Known H, Latent H, $H = 1$, and $H = 4$) in our experiment, giving us 13860 Thompson sampling simulations each model. We then ran the same number of independent simulations for the "Individual" learner (which used only the Gaussian prior described in Appendix E.1).