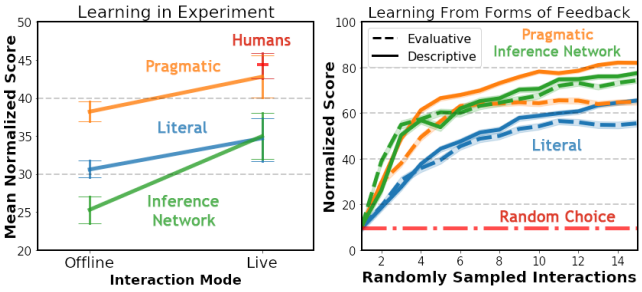

Figure 3: Left: learning within our experiment structure. We plot averaged normalized score over the 10 learning episodes; bars indicate 1 SE (68% CI). Right: learning with specific feedback types. We plot averaged normalized score on 100 test levels after each episode.

teachers with a mean score of 1.09 (mean of all games was 8.53). We augment the data by exchanging the reward function (Fig. 2B), simulating the same episode under a different set of preferences. We take a new reward function and switch both feature counts and token synonyms, preserving the relationships between u^i, τ^i, and w. We repeat this for all 36 possible reward functions, increasing our data volume and allowing us to separate rewards from teachers. We used ten-fold CV with 8-1-1 train-validate-test splits, splitting both teachers and reward functions. Thus the network is trained on one set of rewards (i.e. latent human preferences) and teachers (i.e. linguistic expression of those preferences), then tested against unseen preferences and language. We used stochastic gradient descent with a learning rate of .005 and weight decay of 0.0001, stopping when validation set error increased. We train the network, including embeddings, end-to-end with an L2 loss on the true reward.

Multiple Episodes. Given a (u, τ) tuple, our model predicts the reward $\hat{w}$ associated with every feature. To evaluate it over multiple trials in Section 6, we use a comparable update procedure as our structured models. Concretely, we initialize univariate Gaussian priors over each feature $\mu_0 = 0$, $\sigma_0 = 25$, then run our inference network on each interaction and perform a Bayesian update on each feature using our model's output as an observation with the same fixed noise. For each feature, $P(w^{i+1}|u^i, \tau^i) = \mathcal{N}(\mu_i, \sigma_i) * \mathcal{N}(\hat{w}, \frac{1}{2})$. In all offline testing, we use the network from the appropriate CV fold to ensure it is always evaluated on its test set (teachers and rewards).

6 Results and Analysis

We seek to answer several questions about our models. First, do they work: can they recover a human's reward function? Second, does our sentiment approach provide an advantage over the end-to-end learned model? And finally, do the "pragmatic" augmentations improve the "literal" model? We run a second interactive experiment pairing human teachers with our models and find the answer to all three is yes (Section 6.1). We then analyze how our models learn by testing forms of feedback separately (Section 6.2).

Model	**Experiment**			**Interaction Sampling**			
	Offline	Live	n	All	Eval	Desc	Imp
Literal	30.6	34.7	46	40.5	38.7	40.6	16.7
Pragmatic	**38.2**	**42.8**	47	**52.5**	50.4	**58.2**	**31.7**
Inference	25.3	35.0	55	47.6	**54.3**	53.2	–
Human	–	44.3	104	–	–	–	–

Table 2: Normalized scores averaged over 10 episodes of learning. "Experiment" plays the 10 experiment episodes with a single human; "Interaction Sampling" draws (u, τ) tuples from the entire corpus and plays 100 test levels after each update.

6.1 Learning from Live Interactions

To evaluate our models with live humans, we recruited 148 additional participants from Prolific, an online participant recruitment tool, and paired each with one of three model learners in our task environment. We measured the averaged normalized score across all 10 levels (the mean percentage of the highest possible score achieved). To assess the effect of interactivity, we also evaluated the same three model learners on replayed sequences of (u, τ) tuples from our earlier human-human experiment. The results are shown in Fig. 3 and summarized in Table 2. We conducted a mixed-effects regression (**?**) using performance as the dependent variable, including fixed effects of time (i.e. episode 1, episode 2, etc.), interactivity (i.e. live vs. offline), and learner model (i.e. neural vs. "literal" vs. "pragmatic"), as well as an interaction between interactivity and time. We also included random intercepts and random effects for the learner model for each pair to control for clustered variance. The categorical factor of the learner model was contrast-coded to first compare the neural against the two sentiment models and then compare the two sentiment models directly against each other.

First, we found a significant main effect of time, $t(4138) = 32.77, p < .001$, indicating that performance improves over successive levels. Second, although there was no significant main effect of interactivity, $t(446) = -.08, p = .94$, there was a significant interaction between interactivity and time, $t(4138) = 2.32, p = .02$, suggesting that the benefits of the live condition manifest over successive episodes as the teacher provides feedback conditioned on the learner's behaviors. Finally, turning to the models themselves, we find that the "family" of sentiment models collectively outperform the neural network $t(132) = 3.57, p < .001$ and the "pragmatic" sentiment model outperforms the "literal" one, $t(147) = -2.37, p = .019$. Post-hoc pairwise tests (**?**) find an estimated difference of $d = 7.8, 95\%$ CI: $[2.7, 12.9]$ between the "pragmatic" and "literal" models; $d = -3.77, 95\%$ CI: $[-11.8, 4.3]$ between the neural and "literal"; and $d = -11.5, 95\%$ CI: $[-19.4, -3.7]$ between neural and "pragmatic." This suggests the end-to-end model learns to use most of the literal information in the data, while the inductive biases we encoded into the "pragmatic" model capture additional implicatures.

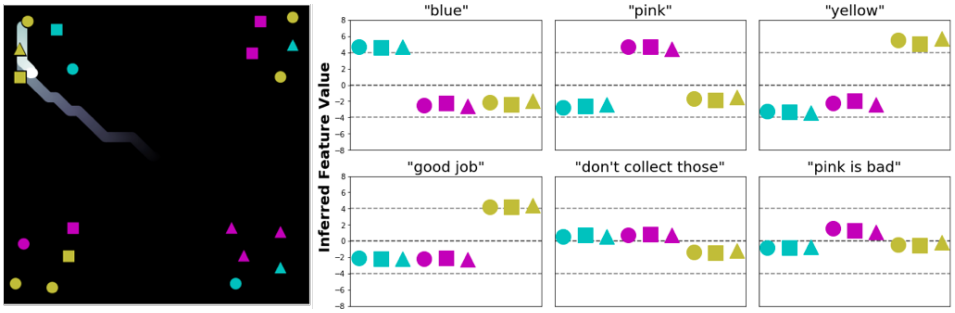

Figure 4: Left: A trajectory from our experiment. Right: Inference network output given this trajectory. Top row: the model learns to map feature-related tokens ("Descriptive" feedback) directly to rewards, independent of the trajectory. Bottom left / center: the model maps praise and criticism ("Evaluative" feedback) through the feature-counts from the trajectory. Bottom right: a failure mode. "Descriptive" feedback with negative sentiment, a rare speech pattern, is not handled correctly.

6.2 Learning from Different Forms of Feedback

To characterize model learning from different "forms" of feedback, we design a second evaluation independent of the experiment structure. Our "episode" sequence is as follows: we draw a (u, τ) tuple at random from the human-human experiment, *update* each model, and have it *act* on our 100 pre-generated test levels. We take its averaged normalized score on these levels. We repeat this procedure 5 times for each cross-validation fold, ensuring the learned model is always tested on its hold-out teachers and rewards. This draws feedback from a variety of teachers and tests learners on a variety of level configurations, giving a picture of overall learning trends. Normalized scores over test levels are shown in Fig. 3 and Table 2 ("Interaction Sampling"). All models improve when learning from the entire corpus ("All") versus individual teachers ("Experiment"). The inference network improves most dramatically, suggesting it may be vulnerable to idiosyncratic communication styles used by individual teachers. We then use our feedback classifier (Section 5.1) to expose models to only a single form of feedback. This reveals that our "pragmatic" augmentations help most on "Descriptive" feedback, which is critical for early learning in the experiment. Finally, we explore our inference network's contextualization process (Fig. 4). It learns to map "Evaluative" feedback through its prior behavior and typical "Descriptive" tokens directly to the appropriate features. We also confirm failure modes on rarer speech patterns, most notably descriptive feedback with negative sentiment. This suggests the learned approach would benefit from more data.

7 Conclusion

We presented two methods to recover latent rewards from naturalistic language: using aspect-based sentiment analysis and learning an end-to-end mapping from utterances and context to rewards. We find that three implementations of these models all learn from live human interactions. The "pragmatic" model in particular achieves near-human performance, highlighting the role of implicature in natural language. We also note that the inference network's performance varies qualitatively across evaluation modes: it outperforms the "literal" model when tested on the whole corpus, but ties it when playing with individual humans ("Interaction Sampling" vs "Experiment - Live"). This underscores the importance of evaluation in realistic interaction settings. We see several future research directions. First, our sentiment models could be improved via theory-of-mind based pragmatics, while our end-to-end approach could benefit from stronger language models (recurrent networks or pre-trained embeddings). Hybridizing sentiment and learned approaches (**??**) could offer the best of both. We also see potential synergies with instruction following: treating commands as "Imperative" feedback could provide a preference-based prior for interpreting future instructions. Finally, we anticipate extending our approach to more complex MDPs in which humans teach both rewards and transition dynamics (**?**). In general, we hope the methods and insights presented here facilitate the adoption of truly natural language as an input for learning.

Acknowledgements

We thank our anonymous reviewers for their thoughtful and constructive feedback. This work was supported by NSF grants #1545126 and #1911835, and grant #61454 from the John Templeton Foundation.

Ethics Statement

Equipping artificial agents with the capacity to learn from linguistic feedback is an important step towards value alignment between humans and machines, with the end goal of supporting beneficial interactions. However, one risk is expanding the set of roles that such agents can play to those requiring significant interaction with humans – roles currently restricted to human agents. As a consequence, certain jobs may be more readily replaced by artificial agents. On the other hand, being able to provide verbal feedback to such agents could expand the group of people able to interact with them, creating new opportunities for people with disabilities or less formal training in computer science.

Modi voluptatibus dolor, vel similique incidunt excepturi quisquam, libero cumque saepe.Eaque iusto inventore quod at error assumenda, quod inventore recusandae quibusdam rem unde harum ipsum.Ea dicta culpa neque aliquid nulla dolore impedit, amet minus beatae sunt rem?