A Related Work

Inverse Reinforcement Learning *Inverse reinforcement learning* (IRL) is a reward learning technique in which the agent infers a reward function given behavioral samples from an optimal policy (Ng & Russell, 2000; Abbeel & Ng, 2004) or a noisy teacher (Ziebart, 2010). It is similar to RLHF in that reward information comes from a teacher rather than the environment, but distinct in that it requires teachers to perform the task well themselves (Milli & Dragan, 2020). RLHF and the HUB framework are most useful in domains such as those presented in Section 5, where the teacher can distinguish good performance, but does not know how to produce it themselves.

Cooperative Inverse Reinforcement Learning *Cooperative inverse reinforcement learning* (CIRL) extends the IRL framework to allow cooperation and collaboration between the agent and the teacher (Hadfield-Menell et al., 2016; Malik et al., 2018). HUB problems can be viewed as a specific class of CIRL games in which there are multiple humans, but they can only act (by providing feedback) when the agent requests it (by querying them). However, CIRL problems are DEC-POMDPS, which are NEXP-complete and thus functionally intractable (Bernstein et al., 2002). By fixing the human policy and arm distributions, the HUB framework reduces the problem to a POMDP with a stationary transition function, which is much more tractable. Optimal agent solutions to the CIRL game balance inference and control to produce several qualitatively valuable behaviors, such as only asking the human questions when necessary (Shah et al., 2020). The algorithm that best solves the HUB problem, ATS, demonstrates similarly conservative querying behavior.

Crowdsourcing Prior work has investigated the related problem of combining feedback from multiple noisy annotators (Dawid & Skene, 1979), often to label training data for supervised learning algorithms. (Raykar et al., 2010) present an approach that learns teacher expertise and uses teacher feedback to fit a classifier simultaneously, while (Rodrigues et al., 2014) generalise gaussian process classification to model noisy annotators and combine their feedback into reliable labels for supervised learning. (Murugesan & Carbonell, 2017) develop a method that also models cost, trading off between querying noisy peer labelers and querying a costly oracle. This body of work underscores the difficulty and importance of combining feedback from varying and noisy teachers in machine learning.

B Theorem 1 Proof

Theorem 1. *If the predicted utility function $\hat{\mathcal{U}}$ and the predicted arm distribution $\hat{\mathcal{D}^{\mathcal{C}}}$ are estimated by executing Algorithm 1 with T samples, then $\hat{\mathcal{U}} \to \mathcal{U}^*$ and $\hat{\mathcal{D}^{\mathcal{C}}} \to \mathcal{D}^{\mathcal{C}*}$ as $T \to \infty$.*

Proof (Sketch). Since the number of arms is finite and they are pulled uniformly as $T \to \infty$, the number of times that a given arm c^k is pulled approaches infinity. Since each pull samples an item from the true distribution $\mathcal{D}^{k*}$ i.i.d., the empirical distribution $\hat{\mathcal{D}^k}$ will approach $\mathcal{D}^{k*}$ in the limit of infinite pulls. This argument applies for all arms $c^k \in \mathcal{C}$, so $\hat{\mathcal{D}^{\mathcal{C}}} \to \mathcal{D}^{\mathcal{C}*}$ as $T \to \infty$. Similarly, in the limit of infinite queries, $\hat{P}(\beta, (i,j))$ will approach $P^*(\beta, (i,j)) = \Pr(i \succ j; \beta, \mathcal{U}^*)$, the true probability that teacher b prefers item i over item j, as determined by Equation 1. Given β, (i,j) and $\hat{P}(\beta, (i,j))$ from the first T timesteps, we can calculate $\Delta_{ij} = \hat{\mathcal{U}}(i) - \hat{\mathcal{U}}(j)$ using Equation **??**. Given $\Delta = [\Delta_{01}, \Delta_{02}, \ldots, \Delta_{NN}]$, u_{max} and u_{min}, we can calculate $\hat{\mathcal{U}}$ as described in Algorithm 1. $\hat{\mathcal{U}} \to \mathcal{U}^*$ as $\hat{P} \to P^*$, which occurs as $T \to \infty$. □

C Theorem 2 Proof

Theorem 2. *Given two items $i, j \in \mathcal{I}$ where $\mathcal{U}(i) < \mathcal{U}(j)$ and the preference probability $P = \Pr(i \succ j; \beta_m, \mathcal{U})$ from Equation 1 we can estimate $\hat{\beta}_m = \frac{1}{z}\beta_m$ as in Equation 3. If Δ_{ij} is known, we can further calculate $\beta_m = z \cdot \hat{\beta}_m$, where $z = -\Delta_{ij}^{-1}$.*

$$\hat{\beta}_m = \ln\Big(\frac{1}{P} - 1\Big). \tag{3}$$

Proof (Sketch). First, we define an affine mapping function $f_{a,b}(x) = ax + b$ such that $f_{a,b}(\mathcal{U}(i)) = 0$ and $f_{a,b}(\mathcal{U}(j)) = 1$. Lemma 3 shows that this is always possible when $\mathcal{U}(i) \neq \mathcal{U}(j)$ and furthermore that $a = \frac{-1}{i-j}$. Let z, y be the parameters that make this mapping for these particular values of $\mathcal{U}(i)$ and $\mathcal{U}(j)$. Note that $z = \frac{-1}{i-j} = -\Delta_{ij}^{-1}$.

Next, suppose we have that $\beta'_m = \frac{1}{a}\beta_m$, it follows that:

$$
\begin{aligned}
P &= \Pr(i^0 \succ i^1; \beta_m, \mathcal{U}) \\
&= \frac{\exp(\beta_m \mathcal{U}(i))}{\exp(\beta_m \mathcal{U}(i)) + \exp(\beta_m \mathcal{U}(j))} && \text{(by Equation 1)} \\
&= \frac{\exp(\frac{\beta_m}{a} \cdot a\mathcal{U}(i) + \frac{\beta_m}{a} b)}{\exp(\frac{\beta_m}{a} \cdot a\mathcal{U}(i) + \frac{\beta_m}{a} b) + \exp(\frac{\beta_m}{a} \cdot a\mathcal{U}(j) + \frac{\beta_m}{a} b)} \\
&= \frac{\exp(\beta'_m \cdot (a\mathcal{U}(i) + b))}{\exp(\beta'_m \cdot (a\mathcal{U}(i) + b)) + \exp(\beta'_m \cdot (a\mathcal{U}(j) + b))} && \text{(by definition of } \beta'_m) \\
&= \frac{\exp(\beta'_m \cdot f_{a,b}(\mathcal{U}(i)))}{\exp(\beta'_m \cdot f_{a,b}(\mathcal{U}(i))) + \exp(\beta'_m \cdot f_{a,b}(\mathcal{U}(j)))} && \text{(by definition of } f_{a,b}) \\
&= \frac{\exp(0)}{\exp(0) + \exp(\beta'_m)} = \frac{1}{1 + \exp(\beta'_m)}.
\end{aligned}
$$

Finally, solving for β'_m yields $\beta'_m = \frac{1}{z}\beta_m = \ln(\frac{1}{P} - 1) \quad \rightarrow \quad \beta_m = z \cdot \ln(\frac{1}{P} - 1)$. □

Lemma 3. *Given any two numbers m, $n \in \mathbb{R}$ such that $m \neq n$, there exists an affine transformation $f_{a,b} : \mathbb{R} \rightarrow \mathbb{R}$ that maps the greater number to 1 and the lesser number to 0.*

Proof (Sketch). Suppose that $m > n$ without loss of generality. We therefore must solve the following system of equations: $f_{a,b}(m) = am + b = 1$ and $f_{a,b}(n) = an + b = 0$. The solution is $a = \frac{-1}{n-m}$ and $b = \frac{m}{n-m} + 1$, which always exists when $m \neq n$. □

D POMCPOW Rollout Policies

ATS solves the HUB-POMDP using *partially observable Monte-Carlo planning with observation widening* (POMCPOW) augmented with a custom rollout policy for estimating the value of leaf nodes in the search tree. We evaluate a *random action* rollout policy, which takes actions uniformly at random from $\mathcal{A} = \mathcal{C} \cup \beta$, a *random arm* rollout policy, which chooses arms uniformly at random from $\mathcal{C}$, and a *best arm* policy, which calculates which arm has the highest expected utility *according to the current belief* b, then always chooses that arm.

Since a utility-maximizing agent will choose arms more often if it believes them to have higher utility, the *best arm* policy rollouts most closely resemble the actions the actual policy would take from belief b, yielding the most accurate value estimates. As a result, ATS with best arm rollouts outperforms the alternatives on the paper recommender domain, as shown in Figure 9. Results are averaged across 25 runs on 20 different paper recommendation tasks.

E HUB Cost effects

We investigate the impacts of teacher query cost on ATS performance by varying professor feedback costs in the paper recommendation domain. We set linear costs $F = \{-1, -2, -3\}$ and scale them by a *cost multiplier*. As in the other paper recommendation experiments, results are averaged across 25 runs on 20 different paper recommendation tasks.

We find that ATS responds rationally to changes in costs, querying teachers more sparingly (Figure 10b) and consequently identifying the best arm more slowly (Figure 10a as overall costs increase. This leads to a slight decrease in overall performance (Figure 10c).

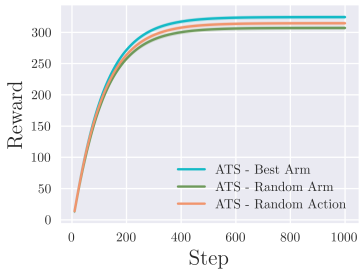
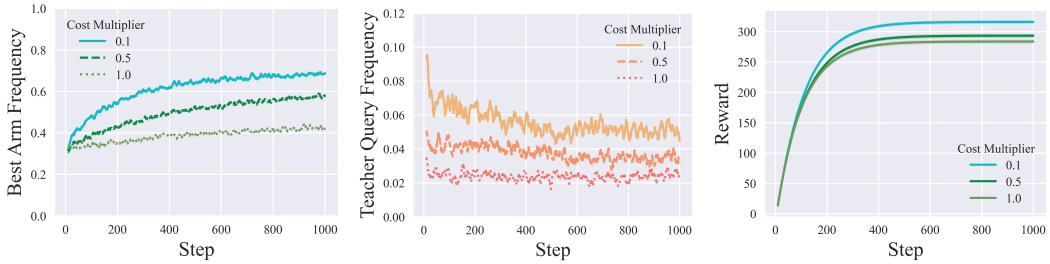


Figure 9: Performance of ATS with various rollout policies. The best arm rollout policy outperforms the random arm and random action rollout policies. All data is averaged across 25 runs on each of 20 HUB problems, smoothed over 10 steps, and discounted with $\gamma = 0.99$.



(a) Frequency of pulling the best arm (b) Frequency of teacher queries (c) Discounted cumulative reward

Figure 10: ATS behavior and performance varies with teacher query costs. Data is averaged across 25 runs on 20 paper recommendation HUB problems and smoothed over 10 steps.