Active Learning within Constrained Environments through Imitation of an Expert Questioner

Kalesha Bullard*, **Yannick Schroecker** and **Sonia Chernova**
Georgia Institute of Technology
{ksbullard, yannickschroecker}@gatech.edu, chernova@cc.gatech.edu

Abstract

Active learning agents typically employ a query selection algorithm which solely considers the agent's learning objectives. However, this may be insufficient in more realistic human domains. This work uses imitation learning to enable an agent in a constrained environment to concurrently reason about *both* its internal learning goals *and* environmental constraints externally imposed, all within its objective function. Experiments are conducted on a concept learning task to test generalization of the proposed algorithm to different environmental conditions and analyze how time and resource constraints impact efficacy of solving the learning problem. Our findings show the environmentally-aware learning agent is able to statistically outperform all other active learners explored under most of the constrained conditions. A key implication is adaptation for active learning agents to more realistic human environments, where constraints are often externally imposed on the learner.

1 Introduction

Active learning (AL) agents are intended to learn from an oracle, often assumed to be human, but typically not *designed* for more realistic human environments. Understanding environmental context however is especially important for robotic agents, generally assumed to be colocated in the environment with the oracle or teacher. Within the robotics community, there has been AL work aimed at understanding [Cakmak *et al.*, 2010; Knox *et al.*, 2013; Gonzalez-Pacheco *et al.*, 2018; Bullard *et al.*, 2018a], modeling [Rosenthal and Veloso, 2011; Racca and Kyrki, 2018], and improving [Chao *et al.*, 2010] interaction with a human partner. An important aspect of the interactive learning problem, this body of work focuses on *interaction* with the teacher, but there still remains the open question of how the learner should integrate reasoning about the *environment* in which it is situated.

Specifically, external constraints imposed on the learner may have *direct* implications for solving the learning problem. For example, a teacher has only a limited time frame of availability or limited cognitive resources that can be devoted to answering the learner's questions. This information may need to influence the learner's questioning policy. However, the problem of trading off learning goals with environmental constraints is relatively unexplored within AL literature, particularly when considering dynamic environments. Yet this problem is important for learning in realistic human settings.

In this work, we investigate the question of how to enable an active learner to reason about its learning objectives within a dynamically changing environment while concurrently considering time and resource constraints provided for solving the learning problem. We use a decision-theoretic approach to active learning, whereby the individual decision criteria (or decision features) within the objective function are hand designed and include both task-centric and environment-centric features. Nonetheless, since the learning agent must consider multiple and diverse decision criteria, it becomes difficult to manually tune the individual objectives. Thus we propose imitation of an expert questioner for learning to weight the decision features. Our approach employs Inverse Reinforcement Learning (IRL) for inferring weights of the objective function from demonstrations of an expert policy.

In the experiments conducted, the agent is given a concept learning problem that it must use active learning to solve, under different environmentally constrained conditions. This work makes the following contributions:

- first AL work to reason about environmental constraints within the objective function of the learner
- first AL work to use imitation learning for mimicking the policy of an expert questioner

We evaluate efficacy using two separate task datasets and show that environmentally-aware reasoning allows our algorithm to significantly outperform an established AL baseline of uncertainty sampling and task-centric questioning strategies examined.

2 Related Work

Active Learning encompasses an extensive body of literature, spanning across several problem domains. We focus here on the most relevant work within the broader space, active learning for robots and embodied artificial agents. Most literature in AL for robots solves learning problems directly relevant

*Contact Author

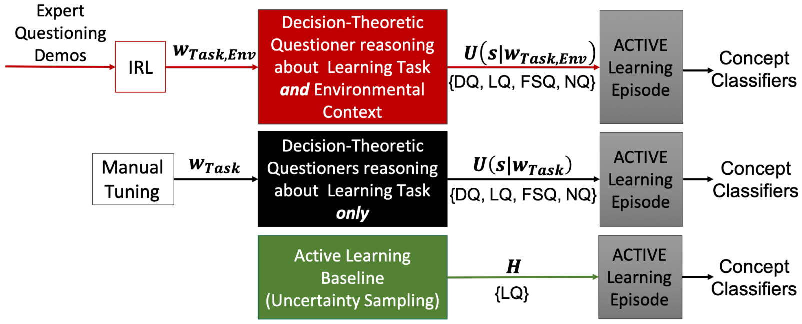


Figure 1: Learning system diagram, illustrating how each active learning strategy performs query selection.

to robotics domains: learning an expert policy to derive desirable robot behavior [Chernova and Veloso, 2009; Lopes *et al.*, 2009; Kroemer *et al.*, 2010; Cakmak and Thomaz, 2012; Daniel *et al.*, 2014; Basu *et al.*, 2018], inferring sequencing constraints on actions in a task [Hayes and Scassellati, 2014], and grounding task-relevant symbols or descriptions [Chao *et al.*, 2010; Kulick *et al.*, 2013; Thomason *et al.*, 2017; Bullard *et al.*, 2018b]. A key limitation however is current approaches *only* reason about the learning problem the agent must solve, but not time and resource constraints imposed by the environment in which the agent is situated. In other words, in prior literature, the learner typically uses an objective function to reason about its learning goals but does not *additionally* consider environmental constraints.

The most relevant prior work explored autonomous arbitration between multiple types of active learning queries, acquiring both feature and instance input from the teacher, and situated in dynamic environments [Bullard *et al.*, 2018b]. A primary contribution of this work was an algorithm for arbitration between diverse query types that could also adapt the frequency of its questioning to the rate at which objects changed in its environment. Nonetheless, a key limitation is the inability to reason about constraints imposed by the teacher or learning environment. Thus, while able to adapt its questioning strategy to the rate of environmental change, the learner has no mechanism for adapting its strategy to the query budget given or amount of learning time allocated.

3 Problem Formulation and Approach

Symbol (or concept) grounding is the problem of mapping symbolic representations to constructs in the physical world [Harnad, 1990]. Specifically, the agent must solve a *task-situated* concept grounding problem, whereby it is given abstract task-relevant concepts to be perceptually grounded in its environment, in a way appropriate for the task. For example, when learning concepts for the *serve breakfast* task, *eggs* scrambled or sunny-side up would be a more appropriate grounding than a dozen cartoned eggs.

We formalize the problem as follows: Given a set of objects $\mathbf{X}$ from a scene in the agent's purview taken at time t, each object instance $\mathbf{x} \in \mathbf{X}$ is represented by a feature vector $\boldsymbol{x}^t = < f_1^t \ldots f_m^t >$. We assume the agent has both exteroceptive and proprioceptive sensors for perceiving its external environment and internal state. Each object instance then is modeled by the superset of features F extracted from the agent's sensors at t (*e.g.* object height or color, position of robot base or end effector). A set of binary classifiers, one for each symbol $y \in Y$, the set of object symbols, each take as input an instance $\mathbf{x}$ and produce a degree of confidence $p(y|\mathbf{x}) = [0,1]$ that $\mathbf{x}$ has label y. For each symbol, a Gaussian Process Classifier was trained. This representation was selected because it both probabilistically models agent uncertainty and learns well from sparse data.

In dynamic environments, groundings also change over time and concept models must be refined accordingly. This may include change in the physical object state (*e.g.* eggs going from being in a shell to scrambled) or objects being replaced within the same category (*e.g.* breakfast beverage being served one day as coffee in a mug and another as orange juice in a glass). Since it is unreasonable to expect a human partner to track the agent's knowledge over time, in a changing environment, we take an *active* concept grounding approach.

3.1 Active Learning for Concept Grounding

Active Learning enables a learner to query an oracle or teacher for information about which is has uncertainty. It typically assumes a query will be made at every turn and seeks to equip the learner with a utility function for selecting an *optimal* query [Settles, 2012]. However, real world environments often do not allow the learner unlimited queries or time for querying, and simultaneously change over time. This means it is not always the best use of time and resources to make a query at every time step, until the query budget is depleted. Thus, employing a traditional AL strategy may not maximize learning in dynamically changing, constrained environments, as shown by [Bullard *et al.*, 2018b].

We present a decision-theoretic AL approach which extends prior work intended for dynamic environments [Bullard *et al.*, 2018b]. In that work, the authors contributed a decision-theoretic framework for arbitrating between multiple types of AL queries, acquiring both informative features and representative training instances from a teacher. Building upon that framework, our approach contributes a model that is able to reason about *both* the agent's concept learning goals *and* external time and resource constraints imposed on the agent. Specifically, the objective function of the learner is *expanded* to include decision criteria which reason about environmental context. Equation 1 shows the learner's objective function used at each turn t to assess the expected utility (EU) of an action a, given the current learning state s_t.

The learning state at t includes: {estimate of posterior probability distributions of $y \in Y$ for all $\mathbf{x} \in \mathbf{X}$, interaction history, query budget, and teaching time allocation}. The set of candidate actions A_t consist of demonstration queries for each of the task-relevant concepts $[DQ(y)\ \forall y \in Y]$, label queries for each object in the current scene $[LQ(\mathbf{x})\ \forall \mathbf{x} \in \mathbf{X}]$, a feature subset query $[FSQ]$ to identify relevant features for discriminating between task concepts, and a no query action $[NQ]$. Thus, there are $|A_t| = |Y| + |\mathbf{X}| + 2$ candidate actions from which the agent can choose at each turn t. Additionally, each of the query types is associated with a cost, given a priori. The learner selects an optimal action a^* as

$$\begin{aligned} a^* &= \underset{a}{\arg\max}\ EU(a|s_t) \\ &= \underset{a}{\arg\max} \sum_{s_{t+1}} P(s_{t+1}|a, s_t)\ \ U(s_{t+1}) \end{aligned} \tag{1}$$

where

$$U(s) = w_1\phi_1(s) + w_2\phi_2(s) + \ldots + w_n\phi_n(s) \tag{2}$$

The set of decision features $\phi \in \Phi$ used in computing $U(s)$ comprise the representation for the agent's objective (decision) function and is primarily what distinguishes prior work from the approaches introduced in this work. $U(s)$ is represented as a function of decision features $\phi : S \to [0,1]^k$, where k is number of decision criteria or individual objectives for which the agent is optimizing.

Baseline Approaches

We employ two AL models from prior literature, as baselines for comparison: a standard uncertainty sampling approach (**U-sampling**) and a state-of-the-art decision-theoretic approach for arbitrating between diverse query actions (**DT-iros**). Uncertainty sampling algorithms are possibly the most commonly employed class of AL strategies in the literature [Settles, 2012; Fu *et al.*, 2013]. They assume a single hypothesis θ and utilize the posterior probability distribution over labels $y \in Y$ given unlabeled instance x, $p_\theta(Y|\mathbf{x})$, in order to detect outliers or instances closest to a decision boundary. Like other standard AL approaches, they query at every turn, each time requesting a label for a maximally informative instance, based upon predetermined selection criteria. A commonly used metric for uncertainty sampling is *prediction entropy*: $-\sum_{y \in Y} p_\theta(y|\mathbf{x}) \log p_\theta(y|\mathbf{x})$. We employ this as our standard AL baseline (**U-sampling**).

Decision-theoretic approaches to active learning simulate all possible outcomes of each candidate query action and optimize with respect to future *expected* utility. This work builds from prior work employing decision theory to arbitrate between diverse types of learning queries, including a supplemental no-query action [Bullard *et al.*, 2018b]. The set of decision features investigated were *average classifier discriminability* and *class distribution uniformity*.

Given a set of instances in the agent's purview $\mathbf{X}$ and a task-relevant concept y, the *classifier discriminability* metric assesses the *range* of probabilities over the set of instances: $p_\theta(y|x_{max}) - p_\theta(y|x_{min})$, where x_{max} and x_{min} are the model's prediction of the most and least probable examples of class y, respectively. Range is a standardized metric of statistical dispersion; an average is taken over all $y \in Y$. Class distribution uniformity assesses selection bias in the training sample, due to an unrepresentative class distribution. It is a useful decision feature in sparse data environments, as has traditionally been the assumption in Learning from Demonstration settings, where the learner does not have sufficient evidence to confidently infer the underlying distribution of classes. This metric incentivizes the learner to minimize sample selection bias. Given this work also seeks to arbitrate between *all* action types, we employ this previously published decision-theoretic objective function, where the number of decision features $k = 2$, as our state-of-the-art baseline (**DT-iros**).

Experimental Approaches

We introduce two experimental questioning policies: a learning-centric model intended to improve the state-of-the-art (**DT-task**) and an environmentally-aware active learner (**DT-task-env**). For the learning-centric model, we propose two *additional* decision features that we believe improves the performance of the originally published DT-iros algorithm, even before consideration of environmental context: *instance variation* and *label prediction margin*, defined by Equations 3 and 4 respectively.

$$IV(s) = \frac{1}{|Y|} \sum_{y \in Y} \frac{\sigma(p(\mathbf{X}|y))}{\mathbb{E}[p(\mathbf{X}|y)]} \tag{3}$$

$$PM(s) = \frac{1}{|X|} \sum_{x \in X} p_{\theta_1}(y_1|\mathbf{x}) - p_{\theta_2}(y_2|\mathbf{x}) \tag{4}$$

Instance variation is a standardized measure of statistical dispersion. Given a class y and a set of scene instances $\mathbf{X}$, it is a measure of relative standard deviation of the class conditional distribution $p_\theta(\mathbf{X}|y)$. Intuitively, it attempts to assess each classifier's ability to recognize variation *amongst* the set of unlabelled instances.

In the context of concept learning, the class-conditional distribution $p(\mathbf{X}|y)$ can be thought of as the likelihood of each unlabelled instance $\mathbf{x} \in \mathbf{X}$ being selected as an example of class y. Given that multiple, diverse instances within a scene may serve as positive examples of a given class, it seems useful to employ decision features which approximate the learner's ability to recognize diversity amongst the set of unlabelled instances in its purview. Because of this, both *classifier discriminability* and *instance variation* are measures of statistical dispersion, but along different dimensions. Whereas, classifier discriminability is a measure of statistical dispersion over the likelihood of instances belonging to a class, instance variation quantifies the statistical dispersion over the features values of instances. The former rewards the learner for differentiating between the most prototypical and improbable examples of each class; the latter rewards the learner for recognizing greater variation between instances. Both decision features incentivize the selection of queries which increase the learner's *recognition* of the underlying *diversity* that exists within the pool of unlabelled instances.

Given an unlabelled instance x and a distribution over class labels $p(\mathbf{Y}|x)$, *label prediction margin* measures the difference between what the learner predicts to be the most probable label y_1 and second most probable label y_2. Previously employed in AL literature [Settles, 2012], it is a measure of uncertainty; as the margin increases, the learner is more confident about its prediction. It is computed for all scene instances, then averaged. This decision feature incentivizes *accuracy* in the class prediction for each unlabelled instance.

Thus, the first decision function proposed in this work (**DT-task**) subsumes the set of decision features considered by DT-iros, considering *four* learning-centric criteria that each optimize for different aspects of the concept learning problem.

The primary contribution of this work however is in the addition of *environmental context* into the AL agent's objective function. We *introduce* the following environmental features:

- *query budget consumption* – measures the proportion of query budget consumed at turn t, given the query history
- *remaining time usage* – measures the proportion of allocated time remaining after turn t
- *non-query time passed* – measures the proportion of consecutive turns no query was made within a sliding time window T_w; here the size of the time window is proportional to the rate of environmental change; it is computed as $t_{NQ} = \frac{n_{NQ}}{|T_w|}$; $t_{NQ} \to 0$ when the learner has just queried and $t_{NQ} \to 1$ when the learner has *not* queried throughout the entire duration of the time window; intuitively this metric is intended to penalize the agent for being *too* passive, in a dynamically changing environment

The environmentally-aware agent's objective function (**DT-task-env**) is then composed of a linear combination of *seven* decision features, a subset of which roughly attempt to estimate progress towards learning goals (*i.e.* learning *task* centric) and the remaining features intended to incentivize wise time and resource management (*i.e. environment* centric). All decision-theoretic learners described can arbitrate between *all* communicative action types.

Given the different types of decision features being considered however, it is challenging to decipher how to trade them off (*e.g.* budget consumption versus prediction margin). One key observation is humans can often intuitively reason about decision criteria that are difficult to compare quantitatively; thus, we propose to observe the strategy of a human expert questioner, given the same learning problem, and infer how the expert trades off the given decision criteria.

3.2 Imitating an Expert Questioning Strategy

Imitation learning seeks to efficiently learn desired behavior by mimicking a domain expert [Osa *et al.*, 2018]. Within imitation learning literature, Inverse Reinforcement Learning (IRL) aims to recover the expert's reward (objective) function from demonstrations of a policy [Ng *et al.*, 2000; Abbeel and Ng, 2004]. We employ a state of the art IRL algorithm, maximum entropy IRL [Ziebart *et al.*, 2008], to infer the weights $w \in W$ of Equation 2, for the active learner's decision features, $\forall \phi \in \Phi$, as wielded by an expert.

The maximum entropy loss function L_{ME} maximizes entropy of distributions over paths followed by the expert, while satisfying the constraint that the learner's decision feature counts should ideally match those of the expert. The problem is formulated as follows: Given $\phi_E(\tau)\ \forall \tau \in T^{demo}$, find an optimal weight vector $\mathbf{w}$ such that

$$\mathbf{w}^* = \arg\max_{\mathbf{w}}\ -\sum_{\tau} p(\tau|\mathbf{w})\ \ln p(\tau|\mathbf{w}) \tag{5}$$

subject to the constraint

$$\mathbb{E}[\phi_E(\tau)] = \mathbb{E}[\phi_L(\tau)] \tag{6}$$

where $\phi_E(\tau)$ and $\phi_L(\tau)$ represent the feature counts of the expert and learner respectively, for a trajectory τ. In our problem domain, a trajectory is a sequence of learning states visited and communicative actions taken $\{s_1, a_1, s_2, a_2, ...s_T, a_T\}$ at each time step $t <= T$, the maximum number of iterations allowed in a learning episode.

Optimization using the maximum entropy loss $L_{ME}(\mathbf{w})$ is equivalent to maximizing the log likelihood of the expert demonstrations [Ziebart *et al.*, 2008; Osa *et al.*, 2018]:

$$\begin{aligned}
\mathbf{w}^* &= \arg\max_{\mathbf{w}}\ L_{ME}(\mathbf{w}) \\
&= \arg\max_{\mathbf{w}}\ \sum_{\tau} p(\tau|\mathbf{w})\ \ln \frac{1}{p(\tau|\mathbf{w})} \\
&\propto \arg\max_{\mathbf{w}}\ \sum_{\tau} p(\tau|\mathbf{w}) \\
\mathbf{w}^* &\propto \arg\max_{\mathbf{w}}\ \sum_{\tau} \ln p(\tau|\mathbf{w})
\end{aligned}$$

Using this formulation, the gradient of the IRL loss, shown in Equation 7, is the difference between the empirical feature counts (demonstrated by the expert) and the expected feature counts, computed from sample trajectories generated with $\mathbf{w}$.

$$\nabla_{\mathbf{w}} L_{ME} = \mathbb{E}_{\pi^{\mathbb{E}}}[\phi_E(\tau)] - \sum_{\tau} p(\tau|\mathbf{w})\phi_L(\tau) \tag{7}$$

We used an empirically determined maximum number of iterations as stopping criteria for the IRL algorithm. Weights for the environmentally-aware active learner's objective function were learned offline and tested for generalization in AL episodes under different environmental conditions.

3.3 Learning Episode

Figure 1 shows the high-level flow for the learning system. For each active learning episode conducted, the task-relevant concepts and questioning strategy are given as input. Within an episode, at each turn t, the agent perceives all objects in its purview, computes its estimate of the posterior probability distributions $p(y|x)\ \forall x, y$ to update learning state s_t, determines the set of candidate actions A_t, computes $EU(a|s_t)\ \forall a \in A_t$, then takes an optimal action a^*. The learning episode concludes once $t = T$.

4 Evaluation

This work explores an AL strategy *designed* to optimize for environmental constraints and proposes an imitation learning approach for accomplishing this. Toward this end, we test two hypotheses: (1) Reasoning *additionally* about environmental context can enable an AL agent to adapt its questioning strategy and improve its learning performance under constrained conditions, and (2) Imitation Learning can be used

(a) Pasta (*state*:in box) (b) Pasta (*state*:in pot) (c) Pasta (*state*:in bowl)

(d) Banana (*state*:bunch) (e) Banana (*state*:single) (f) Banana (*state*:sliced)

Figure 2: Illustration of object state changes for *main dish* and *fruit* objects classes in *prepare-lunch* task.

to infer an expert's strategy for managing time and resources allocated to solve a given learning problem, then generalized to other constrained environments. As illustrated in Figure 1, each of the questioning strategies conduct their own learning episodes, during which binary classifiers are trained for all task-relevant classes, based upon information gathered. We use recognition accuracy on hold-out test sets for assessing the concept models learned, given each questioning strategy.

4.1 Experimental Design

In evaluating the AL approaches, we focus experiments on a concept grounding task in a dynamic environment, under *different* environmentally constrained conditions. We also examine performance on another task for generalization of the learned decision feature weights across tasks. Both concept grounding tasks are given the same four abstract concepts to ground (main dish, snack, fruit, and beverage), but are generated from different object RGB-D datasets and represent different properties of dynamic change.

The *prepare-lunch* task, the most difficult of the two learning problems, is our focus; it places emphasis on the same objects changing *state*, as one might expect over the course of the task (*e.g.* pasta going from being in a box in the pantry to being cooked in a pot to being served in bowl for lunch.) It was extracted from a local RGB-D object dataset focused on state-change. Figure 2 shows an example of the type of dynamic change the learner may expect to see in this task. In the second *pack lunchbox* task, objects do not change state, but have greater within-category diversity. For example, the *fruit* class contains apples, oranges, peaches, and pears, and the *beverage* class contains varieties of both soda and water. This task was extracted from the University of Washington RGB-D dataset of common household objects [Lai *et al.*, 2011]. As both tasks are from prior literature, details regarding data collection can be found in [Bullard *et al.*, 2018b].

Since RGB-D datasets were being used for evaluation, in order to create a learning environment that more closely approximates real-world settings, we simulated multi-modal features, representing features extracted from a robot's other sensors[1]. Gaussian noise was added to all simulated features since robot sensor data is typically noisy. We also simulated dynamic change in the environment by sampling a new set of object images at a predetermined rate, to represent the scene changing. At each turn t, O contains only one observation (image) of each object in the scene. To simulate environmental change, the perceptual system generates a new set of observations. Else, it outputs the set of observations from $t-1$. In the prepare-lunch task, objects can change state.

Using the complete RGB-D object datasets, we generated five smaller *task* training data samples and one disjoint hold-out test sample for each task. Since the UW dataset is several orders of magnitude larger than the local dataset created, the data sample sizes vary by task. Each of the training and test task samples are 80 images and 40 images respectively for the prepare-lunch task and 3200 images and 800 images respectively for the pack-lunchbox task.

There are four AL algorithms being evaluated on each task: the uncertainty sampling baseline (**U-sampling**), two task-centric decision-theoretic learners (**DT-iros** and **DT-task**), and our experimental environmentally-aware decision-theoretic approach (**DT-task-env**), which reasons about *both* learning objectives *and* environmental constraints.

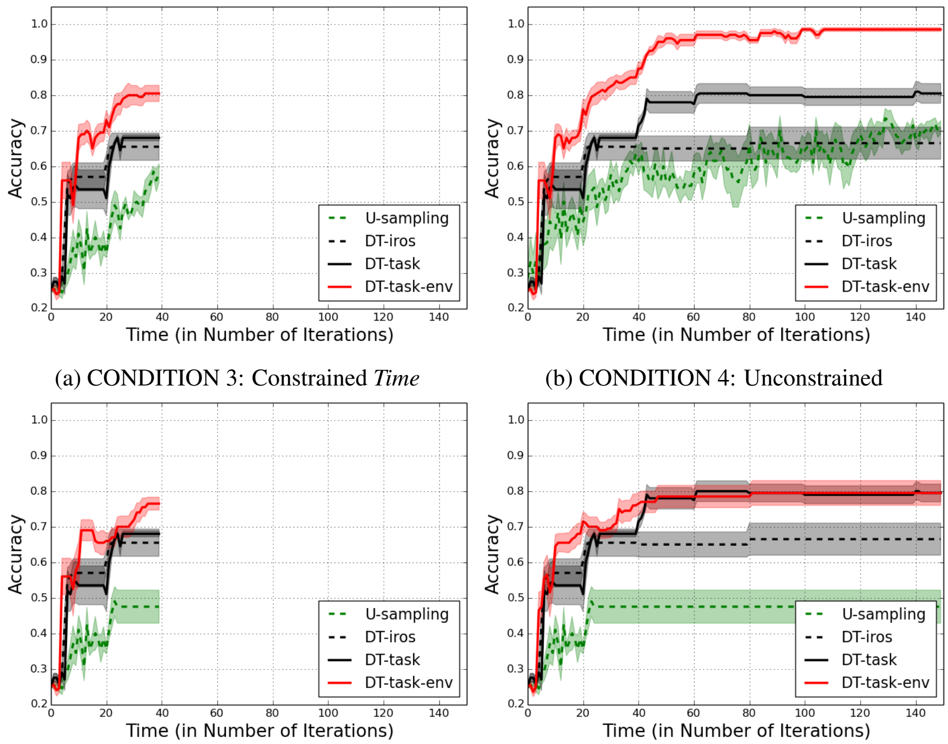
For training of DT-task-env decision feature weights, an expert questioner was given a very constrained query budget (15) and time allocation (30 turns) to ground the *prepare-lunch* task concepts. During training, the environmental scene changed every 10 turns, and major object state changes took place with each scene change; thus spreading the query budget out over the allocated time period affords the opportunity to acquire a more diverse and representative training sample. This was a key part of the strategy employed by the expert used, which was one of the authors of this paper.

The expert provided three demonstrations of questioning sessions (learning episodes) in the training scenario. As part of the strategy demonstrated, the expert also always requested relevant features for discriminating between concepts (FSQ) early in the learning episode (within the first five turns), focused most queries on the least costly query type, and focused on quickly acquiring representative training examples for each class. During IRL training, the maximum number of iterations was set to 100, and we selected the set of weights $\mathbf{w}^*$ that performed best on the validation set. Qualitatively examining the rollouts associated with $\mathbf{w}^*$ under the training conditions, the behavior of the imitation learner was able to closely match the expert's questioning strategy. Environmental conditions were held constant across demonstrations and IRL training. Values changed for testing were time and budget allocation and frequency of environmental change.

4.2 Results

To test our hypotheses, we first investigated generalization to *different* environmentally constrained conditions. The goal was to vary time on one axis and resources (query budget) on the other axis. In order to have a feasible stopping point, we

[1] *object's location relative to interest points in the environment (e.g. counter top, stove, refrigerator, pantry), the object's location relative to the robot base, absolute location of robot's base in the environment, location of the robot's base with respect to the counter top, the robot's joint positions for each arm, pose of the robot's hands, robot's hand states (open vs closed), weight of the object, and max/min/average volume of noise in the environment over duration of learning episode*



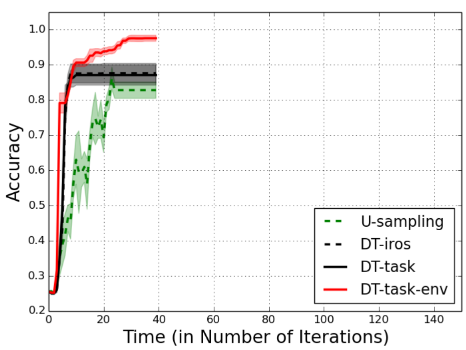
(a) CONDITION 3: Constrained *Time*

(b) CONDITION 4: Unconstrained

(c) CONDITION 1: Constrained *Time* & *Budget*

(d) CONDITION 2: Constrained *Budget*

Figure 3: Prepare Lunch Task. Shows performance (test accuracy with standard error) for each AL strategy under different environmentally constrained conditions. Parameters of allocated time and query budget imposed on the learner vary, with: (a) *only time* constrained [budget: high (500), time: low (40)], (b) *neither* time nor query budget constrained [budget: high (500), time: high (150)], (c) *both* time and query budget constrained [budget: low (25), time: low (40)], and (d) *only* query *budget* constrained [budget: low (25), time: high (150)].

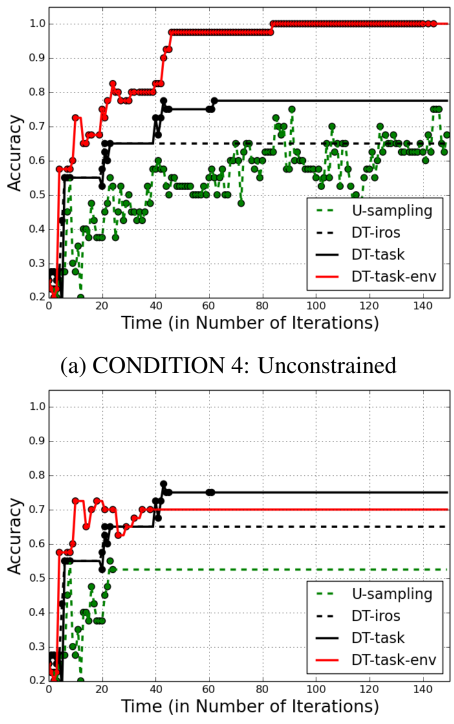


(a) CONDITION 1: Constrained *Time* & *Budget*

Figure 4: Pack Lunchbox Task. Shows performance (test accuracy with standard error) on a separate task, under the most constrained experimental condition: *both* time *and* query budget constrained.

could not truly allow unlimited time or query budget; however, as defined here, the *constrained* versus *unconstrained* parameters denote an order of magnitude difference in allocation. All action types were assigned an a-priori cost, which is 2 for demo queries and feature subset queries, 1 for label queries, and 0 for no query. This can be assigned in any way desired, but for our purposes, was intended to map roughly with the cognitive load required by the teacher in answering a particular type of question. Figure 3 shows learning curves for each combination of time and resource parameters. For each task, learning curves are averaged over 5 runs, each run sampling from a different pre-generated task training data sample. Examining the subfigures: from left to right, time allocation is increased (from 40 to 150) and from bottom to top, query budget is increased (from 25 to 500). Thus fig 3c is the most constrained (budget 25, time 40) and most similar to the training scenario (budget 15, time 30). Overall, the left half roughly corresponds to the agent being given approximately 20 questions, assuming the most costly queries are minimized, whereas the right half corresponds to the agent being given unlimited queries during the time allocated. In testing, once a strategy has exceeded its query budget, it is no longer allowed to make queries, representing complete resource consumption. Thus for the remainder of the episode, it must select the no-query action at no cost. It should also be stated that we assume at most one query per turn.

We used the Mann-Whitney U-test to perform pairwise statistical comparisons of our experimental approach (DT-task-env) with each of the baseline approaches at the end of the allocated learning time, for all four experimental conditions. A one-tailed test was conducted, as the goal was to understand if the experimental environmentally-aware approach *improves* performance over the baseline task-centric approaches to AL. In conditions 1, 3, and 4, DT-task-env statistically outperforms *all* other approaches by the end of the learning time. In



(a) CONDITION 4: Unconstrained

(b) CONDITION 2: Constrained *Budget*

Figure 5: Questioning Behavior of each Strategy in Prepare-Lunch Task for *one* training sample. Dots indicate when a query is made.

condition 2, it statistically outperforms all approaches except DT-task, compared to which it performs equivalently. Using the second (pack-lunchbox) task, we were able to replicate our results in another domain. We only show the most constrained condition (limited time *and* query budget) in Figure 4. Overall, there is a clear pattern. DT-task-env *always* performs *at least* as well as the task-centric baselines but moreover *dominates* task-centric approaches under most of the environmental conditions examined. This confirms our second hypothesis that a questioning strategy learned through imitation of an expert in one environment *can* be used to generalize to other constrained environments. Our first hypothesis however is *not* supported by findings from Condition 2.

To better understand what behavior leads to these findings, we analyze learning episodes from one training data sample under two different experimental conditions in Figure 5. The dots represent points where queries were made for the given strategy. We find that given a limited query budget and ample time (figure 5b), DT-task and DT-task-env employ very similar *conservative* strategies. Both use most of their budget closer to the beginning of the episode, as they attempt to build initial concept models, but also attempt to modulate budget consumption with rate of environmental change. However, when the agent is allowed to ask unlimited queries given the same time frame (figure 5a), these two strategies behave very differently. Whereas the task-centric learners employ exactly the same strategy (because they have no ability to reason about environmental constraints), DT-task-env employs a very *liberal* strategy. In fact, it makes at least an order of magnitude more queries both than DT-task and DT-iros (134 versus 22 and 9), largely accounting for its complete domination over the other strategies. Also notably, U-sampling asks a question at *every* turn until it exceeds its budget, since it is able to modulate *neither* for environmental change *nor* for external constraints. It also does not have the capability to autonomously select a feature subset query, so it must rely upon computational feature selection for solving its learning problem. The decision-theoretic approaches, by contrast, are able to reason about and request an FSQ early in the episode, making them significantly more sample efficient.

The key implication of all of the experimental findings is that the DT-task-env strategy has the ability to effectively *adapt* its questioning behavior *both* to the rate of environmental change (like DT-iros and DT-task) *and* to time and resource constraints imposed externally. In more realistic environments, this is compelling as it gives human partners the capability to specify their own time and cognitive load constraints, with the understanding that the agent can integrate this knowledge into its reasoning about the learning task.

5 Discussion

This work contributes a new cognitive capability for active learners in more realistic contexts, by enabling them with a policy to trade off learning objectives with environmental constraints. Yet, it is not without its limitations. The properties of environmental change captured by the object datasets used, serve only as a proxy for partial observability encountered in the real world. In realistic environments, scenes are often cluttered and scene change happens continuously.

Additionally, the active learning models contributed are intended for use by any type of artificial agent and thus agnostic to an agent's embodiment. However, *what* an agent decides to ask cannot always be decoupled from *how* it must use its embodiment to execute the query. For example, query cost may increase if a query requires fine-grained manipulation that is difficult for the agent to maneuver or simply takes a longer time to generate than other queries. Ideally, this should be incorporated into the agent's decision function.

Future work could explore how to adapt these methods, given more complex perceptual data or agent embodiment as input for the questioning framework.

6 Conclusion

This work contributed a first exploration and novel computational approach for solving the problem of active learning under externally imposed time and resource constraints. Imitation of an expert questioner's policy was used to learn to weight the diverse set of decision criteria for an environmentally-aware active learner. Experiments were conducted under various environmentally constrained conditions and on two concept learning tasks. Key findings show the experimental approach presented statistically outperformed a standard uncertainty sampling baseline and the strictly task-centric active learners under most environmental conditions; thus representing a promising alternative for active learners in more realistic human environments.

References

[Abbeel and Ng, 2004] Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 1. ACM, 2004.

[Basu *et al.*, 2018] Chandrayee Basu, Mukesh Singhal, and Anca D Dragan. Learning from richer human guidance: Augmenting comparison-based learning with feature queries. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, pages 132–140. ACM, 2018.

[Bullard *et al.*, 2018a] Kalesha Bullard, Sonia Chernova, and Andrea L Thomaz. Human-driven feature selection for a robotic agent learning classification tasks from demonstration. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6923–6930. IEEE, 2018.

[Bullard *et al.*, 2018b] Kalesha Bullard, Andrea L Thomaz, and Sonia Chernova. Towards intelligent arbitration of diverse active learning queries. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6049–6056. IEEE, 2018.

[Cakmak and Thomaz, 2012] Maya Cakmak and Andrea L Thomaz. Designing robot learners that ask good questions. In *ACM/IEEE Int Conf on Human-Robot Interaction*, pages 17–24, 2012.

[Cakmak *et al.*, 2010] Maya Cakmak, Crystal Chao, and Andrea L Thomaz. Designing interactions for robot active learners. *Autonomous Mental Development, IEEE Transactions on*, 2(2):108–118, 2010.

[Chao *et al.*, 2010] Crystal Chao, Maya Cakmak, and Andrea Lockerd Thomaz. Transparent active learning for robots. In *ACM/IEEE Int. Conf. on Human-Robot Interaction*, pages 317–324, 2010.

[Chernova and Veloso, 2009] Sonia Chernova and Manuela Veloso. Interactive policy learning through confidence-based autonomy. *Journal of Artificial Intelligence Research*, 34(1):1, 2009.

[Daniel *et al.*, 2014] Christian Daniel, Malte Viering, Jan Metz, Oliver Kroemer, and Jan Peters. Active reward learning. In *Robotics: Science and Systems*, 2014.

[Fu *et al.*, 2013] Yifan Fu, Xingquan Zhu, and Bin Li. A survey on instance selection for active learning. *Knowledge and information systems*, pages 1–35, 2013.

[Gonzalez-Pacheco *et al.*, 2018] Víctor Gonzalez-Pacheco, María Malfaz, A Castro-Gonzalez, José Carlos Castillo, F Alonso, and Miguel Angel Salichs. Analyzing the impact of different feature queries in active learning for social robots. *International Journal of Social Robotics*, pages 1–14, 2018.

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

[Hayes and Scassellati, 2014] Bradley Hayes and Brian Scassellati. Discovering task constraints through observation and active learning. In *2014 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 4442–4449, 2014.

[Knox *et al.*, 2013] W Bradley Knox, Peter Stone, and Cynthia Breazeal. Training a robot via human feedback: A case study. In *International Conference on Social Robotics*, pages 460–470. Springer, 2013.

[Kroemer *et al.*, 2010] OB Kroemer, Renaud Detry, Justus Piater, and Jan Peters. Combining active learning and reactive control for robot grasping. *Robotics and Autonomous Systems*, 58(9):1105–1116, 2010.

[Kulick *et al.*, 2013] Johannes Kulick, Marc Toussaint, Tobias Lang, and Manuel Lopes. Active learning for teaching a robot grounded relational symbols. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 1451–1457. AAAI Press, 2013.

[Lai *et al.*, 2011] Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *IEEE Int. Conf. on Robotics and Automation*, pages 1817–1824, 2011.

[Lopes *et al.*, 2009] Manuel Lopes, Francisco Melo, and Luis Montesano. Active learning for reward estimation in inverse reinforcement learning. In *Machine Learning and Knowledge Discovery in Databases*, pages 31–46. Springer, 2009.

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

[Osa *et al.*, 2018] Takayuki Osa, Joni Pajarinen, Gerhard Neumann, J Andrew Bagnell, Pieter Abbeel, Jan Peters, et al. An algorithmic perspective on imitation learning. *Foundations and Trends® in Robotics*, 7(1-2):1–179, 2018.

[Racca and Kyrki, 2018] Mattia Racca and Ville Kyrki. Active robot learning for temporal task models. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, pages 123–131. ACM, 2018.

[Rosenthal and Veloso, 2011] Stephanie Rosenthal and Manuela Veloso. Modeling humans as observation providers using pomdps. In *RO-MAN, 2011 IEEE*, pages 53–58. IEEE, 2011.

[Settles, 2012] Burr Settles. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1):1–114, 2012.

[Thomason *et al.*, 2017] Jesse Thomason, Aishwarya Padmakumar, Jivko Sinapov, Justin Hart, Peter Stone, and Raymond J Mooney. Opportunistic active learning for grounding natural language descriptions. In *Conference on Robot Learning*, pages 67–76, 2017.

[Ziebart *et al.*, 2008] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. In *AAAI*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.