Understanding the Relationship between Interactions and Outcomes in Human-in-the-Loop Machine Learning

Yuchen Cui\textsuperscript{1}\textsuperscript{*}, Pallavi Koppol\textsuperscript{2}\textsuperscript{*}, Henny Admoni\textsuperscript{2}, Scott Niekum\textsuperscript{1}, Reid Simmons\textsuperscript{2}, Aaron Steinfeld\textsuperscript{2} and Tesca Fitzgerald\textsuperscript{2}†

\textsuperscript{1}Department of Computer Science, The University of Texas at Austin
\textsuperscript{2}School of Computer Science, Carnegie Mellon University

Abstract

Human-in-the-loop Machine Learning (HIL-ML) is a widely adopted paradigm for instilling human knowledge in autonomous agents. Many design choices influence the efficiency and effectiveness of such interactive learning processes, particularly the interaction type through which the human teacher may provide feedback. While different interaction types (demonstrations, preferences, etc.) have been proposed and evaluated in the HIL-ML literature, there has been little discussion of how these compare or how they should be selected to best address a particular learning problem. In this survey, we propose an organizing principle for HIL-ML that provides a way to analyze the effects of interaction types on human performance and training data. We also identify open problems in understanding the effects of interaction types.

1 Introduction

Human-in-the-loop machine learning (HIL-ML) [Fails and Olsen Jr, 2003; Amershi et al., 2014] describes learning processes in which an agent learns from human interaction to acquire data for improving its performance. There has been a recent increase in the number of interaction types through which a teacher may provide training data, such as providing a demonstration, indicating a preference between two possible actions the agent may take, correcting the agent’s actions, or providing critiques for the agent’s trajectories. To build effective HIL-ML systems, it is important to understand how interaction type interplays with other components of a HIL-ML pipeline to eventually affect the system’s learning outcomes. For example, performance of a machine learning model is often bounded by the training data’s quantity [Kalapanidas et al., 2003; Halevy et al., 2009] and quality [Cortes et al., 1994; Hänsch and Hellwich, 2019]. Additionally, studies in cognitive science and human-robot interaction have shown that human factors, such as mental workload and perceived usability, affect people’s performance on tasks [Longo, 2018; Haapalainen et al., 2010]. In this paper, we survey existing work on HIL-ML through the lens of interaction types. We organize this review by the relationships between interaction type, human performance, and training data in order to underscore the effects of interaction type on learning outcomes.

1.1 Scope and Contributions

A significant challenge in designing HIL-ML systems is their interconnected nature: the agent’s behavior when querying the teacher may affect the teacher’s response, which in turn affects the training data that informs the agent’s future behavior. As a result, HIL-ML is a very broad area of research that lies at the intersection of computer science, cognitive science, and psychology. To the best of our knowledge, this survey is the first to both formalize a comprehensive model of the HIL-ML paradigm and situate prior and ongoing research regarding how the interaction type can affect both a human’s teaching performance and the agent’s consequential learning outcomes.

The main contributions of this paper are:

• A relationship graph for understanding the role of interaction types in HIL-ML systems, integrating and summarizing insights from studies in both machine learning and human factors.

• Surveys of recent papers in relevant fields that support the proposed relationship graph.

• Open problems for future research, particularly open questions to which the answers would support researchers in robustly comparing and analyzing learning interactions for HIL-ML systems.

Existing research has investigated the impact of individual interaction types on learning outcomes. Jeon et al. [2020] proposed a framework unifying different interaction types in the reward learning literature and compared how different interaction types influence learning of a reward function assuming optimal inputs. Recent work of Koppol et al. [2021] identified differentiating features between interaction types and investigated how these features influence human factors when users are asked to provide training data. Each of these prior works presents one way in which interaction types ultimately affect the HIL-ML process; our work unifies these lenses into
a comprehensive view of the influence of interaction types in the overarching HIL-ML process.

The specific domain and physical interface in which an HIL-ML system is deployed will affect the system’s learning outcome and the human teacher’s experience [Koppol et al., 2021; Ke et al., 2020; Dudley and Kristensson, 2018]. Rather than attempt to enumerate all possible effects of domain and interface design, we focus on features of interaction types that differentiate them at the algorithmic level. As a result, the design choices and relationships analyzed in this paper may be considered within any domain. To provide concrete examples, however, we ground our discussion in the context of a warehouse robot tasked with restocking items. In this example, the robot collects data from humans to train separate models for object recognition and manipulation.

2 Relationship Graph

In HIL-ML systems, a primary goal of selecting an interaction type is to choose the one that best aligns the learning objectives and learning objectives of the overall system. These learning outcomes are dependent on the training data obtained by the system. To understand how interaction type influences learning outcomes, it is important to understand its relationship with training data. Existing studies show that interaction type influences training data in at least two different ways. First, the interaction type directly determines the form of label that is collected in the training data, as well as the training implications of that label (e.g., how that label should be converted into a reward update [Jeon et al., 2020]). Second, the interaction type also influences human performance on the teaching task [Koppol et al., 2021], which can affect the quantity and quality of training data. We propose a relationship graph (Fig. 1) as an organizing principle for HIL-ML systems in order to comprehensively analyze the effects of interaction types. We define each node of the graph in the remainder of this section. Later, in Sections 3-5, we explore the relationships indicated by the graph edges.

2.1 Problem Specification

The problem specification consists of the objectives and constraints of a given learning problem. Learning objectives describe the goals of designing a HIL-ML system, which may consist of objectives such as the expected performance on the training, testing, and generalization datasets, output consistency, sample efficiency, (adversarial) robustness [Zhang et al., 2019a], and/or explainability [Rosenfeld and Richardson, 2019]. Constraints of a HIL-ML system specify the requirements and limitations, which may include the size of the training data set, physical limits, safety requirements, and so forth.

In the restocking robot domain, for example, the learning objective of the robot’s object recognition model is to meet an expected object detection accuracy above some threshold, while also being robust to changes in lighting conditions in the warehouse. Within the robot’s manipulation model, the objective is to generalize its grasping model learned from a small set of objects to robustly grasp novel ones, under the constraint that collisions in any form should be avoided.

2.2 Interaction Type

The interaction between humans and learning agents can take many forms. Cakmak and Thomaz [2012] proposed a categorization of interaction queries that correspond to questions that people tend to ask: label, demonstration, and feature queries. Zhang et al. [2019b] presented a survey on different types of human guidance specifically for deep reinforcement learning and identified four different learning scenarios: standard imitation learning, learning from evaluative feedback, imitation from observation, and learning attention from human. Najar and Chetouani [2021] presented a taxonomy of “advice” for RL agents, categorizing it first according to whether it provides contextual or general advice, and then according to whether it indicates feedback, instructions, or constraints. Recent work of Koppol et al. [2021] identified four archetypes of interactions in the literature that differ by the amount of data the learner requests feedback on, the amount of data the teacher provides in their response, the granularity of the teacher’s response, and the responses the teacher can choose from. These archetypes are: Showing, Categorizing, Sorting, and Evaluating. We will use these as canonical categories of interactions in the remainder of our survey, and ground each in a specific scenario: the restocking robot is learning where to place new items on the shelf.

Showing. The teacher provides a demonstration of the agent’s expected output. This form of interaction is common in the highly-active research field on Learning from Demonstration [Argall et al., 2009; Chernova and Thomaz, 2014]. For the restocking robot, this feedback is provided as an image-label pairing for each object to be shelved, and/or a series of example trajectories demonstrating where to shelve each object. Alternatively, the teacher may verbally explain the expected behavior in the form of “advice” indicating what the agent should or should not do in a particular state [Krenning et al., 2016; Najar and Chetouani, 2021].
**Categorizing.** The teacher provides one or more labels from a predefined set. For the restocking robot, this may involve the teacher watching the robot place an object onto a shelf, and then providing a single rating of the robot’s performance [Daniel et al., 2014]. Or, the teacher may indicate which object (from a set of candidate objects) the robot should place on the shelf [Fitzgerald et al., 2018].

**Sorting.** The teacher indicates their relative preferences over a set of choices presented by the agent [Sadigh et al., 2017]. For example, the restocking robot may suggest n candidate locations for a particular item, which the teacher then orders by their appropriateness. The restocking robot could also suggest two potential grasps for an object and then ask the teacher for their preference.

**Evaluating.** The teacher provides granular feedback on an agent’s proposed or executed actions. For example, the teacher may supervise the restocking robot and correct its behavior in anticipation of an error (e.g., adjusting the robot’s grasp of an object if they believe it to be unstable). These corrections may range from fine-tuned adjustments of the robot’s end-effector pose [Argall et al., 2010; Fitzgerald et al., 2019] to perturbations of the robot’s intended trajectory [Bajcsy et al., 2017] to changes in the hierarchical structure of the task [Gutierrez et al., 2018].

### 2.3 Human Performance

In a HIL-ML system, the agent’s task is to achieve specified objectives, and the human teacher’s task is to provide data that supports the learner. The quality of this training data is critical to the agent’s learning outcomes, and is affected by how well the human teacher executes their teaching task. We define human performance as a human’s ability to provide accurate feedback during a learning interaction. For example, if the restocking robot requests feedback on grasping a new object, human performance consists of the human’s ability to provide a demonstration, correction, or other indicator that results in a stable grasp of the object. The teacher’s ability to provide quality data depends on how they may provide feedback; if the robot requests a ranking between two candidate grasping poses that are equally bad, the teacher may have difficulty deciding between the two and is unable to express feedback about how the robot should grasp the object.

### 2.4 Training Data

Training data is the set of data samples generated by the human teacher through interacting with the learning agent. For the restocking robot, this could consist of object type labels, robot arm trajectories, and/or ratings of trajectories.

### 2.5 Learning Outcomes

The outcomes of HIL-ML systems are the objective measures of performance of the trained system. In an effective HIL-ML system, these outcomes should fulfill the previously-described learning objectives that define the performance goals of the system. We first consider three common performance-based learning outcomes. Training performance reflects the model’s ability to represent its training data. The exact performance metric is domain-specific. In supervised learning, training accuracy is a common metric representing how well the trained model can reproduce the expected output from its training dataset. For reward-based task learning, policy loss in the training environment is often used as a performance metric. Testing performance reflects the model’s ability to produce the expected output for inputs that are drawn from the same distribution as, but not included in, the original training dataset. This testing dataset represents the set of problems that the trained model is expected to encounter in its domain. Generalization performance reflects the model’s ability to produce the expected output for inputs that are drawn from a significantly different distribution from the original training dataset. This type of learning outcome may not apply to all learning domains, but is frequently used in domains where the agent learns multiple tasks (e.g. one-/few-shot learning).

In the restocking robot example, the agent’s learning outcome for the object recognition task would be its training and testing performance in classifying catalogued items, and generalization performance in identifying newly-introduced, uncatalogued objects.

### 2.6 Relationships

Having introduced the nodes of our proposed relationship graph (Fig. 1), we now define the edges in the graph through surveying relevant literature. Throughout this survey, we identify prior work highlighting the importance of the following relationships within HIL-ML systems that are affected by interaction types:

(i) Data implication: characteristics of training data that affect the agent’s ability to fulfill its learning outcomes,

(ii) Feedback interpretation: how the teacher’s feedback is synthesized into training data,

(iii) Teaching quality: how the human teacher’s experience affects the quality of their feedback, and

(iv) User experience: how the queries posed via this interaction type are perceived by the human teacher.

We subsequently address each of these relationships.

### 3 Data Implication

The quality and quantity of training data affect learning outcomes in various ways. Two measures of quality of a data set are noisiness and distribution. Quantity straightforwardly refers to the number of samples in the data—however, how much data is enough is often determined by the complexity of the task itself and the learning objectives of the agent.

**Noise.** Noise can be introduced during data collection through human error (labeling error). Noisy data often leads to a false measure of training and testing performance [Cortes et al., 1994; H änsch and Hellwich, 2019], the effect of which is specific to the training algorithm [Kalapanidas et al., 2003]. During data collection, especially when crowdsourcing [Lease, 2011], it is important to control such noise in labels through explicitly correcting for labeling bias [Snow et al., 2008] or modeling noisy labelers [Sheng et al., 2008].
Distribution. This encompasses the diversity, biases, and representativeness of the data sets. ML models may overfit not only to training data, but also to test and generalization data as a result of the research community using identical benchmarks [Recht et al., 2019]. It is important to design distributions of test and generalization sets that account for domain shifts to better understand generalization errors of ML models [Subbaswamy and Saria, 2020].

Quantity. Data quantity has proven to be a crucial factor of performance of ML models [Halevy et al., 2009], especially for deep neural networks, as demonstrated on image classification [Deng et al., 2009] and natural language processing [Brown et al., 2020]. Small training datasets can lead to overfitting [Raudys et al., 1991]. Performance of deep models on vision tasks increases logarithmically with the volume of data as a result of the research community using identical benchmarks [Recht et al., 2019].

4 Feedback Interpretation

Information gain has been used to select queries in active HIL-ML systems in studies of individual interaction types, such as when querying a teacher to critique a robot’s motion [Cui and Niekum, 2018] or when querying a teacher to indicate their preference over two proposed actions [Biyik et al., 2020]. Recent work by Jeon et al. [2020] proposes a formulation of information gain for finding the best feedback type for reward learning, assuming optimal feedback. We build on these studies and formulate information gain for measuring the effect of interaction types on training data for HIL-ML systems. Information gain represents the expected change in the model’s information entropy (H) resulting from new information. In a HIL-ML context, this information consists of the training data obtained from one interaction between the teacher and agent. In the remainder of this section, we frame this training data as resulting from a choice made by the human teacher in response to the agent’s query. Each interaction type defines the ways in which the agent may query the teacher for training data and, as a result, defines the number and distribution of possible responses by the human teacher to the agent’s query. We ground the problem of calculating the information gain $\Phi_i$ for the optimal query of interaction type $i$ as follows:

$$\Phi_i(\omega, s) = \max_{q \in Q_i(\omega, s)} IG(\omega, C_i(q))$$  

$$= H(\omega) - \min_{q \in Q_i(\omega, s)} \mathbb{E}_{c \in C_i(q)}[H(\omega|c)]$$

where $s$ is the state in which the query $q$ occurs using interaction type $i$, and $\omega$ represents the random variable for model weights/parameters. This formulation also relies on a function $Q$ that produces a set of queries, and a function $C$ that produces a set of feedback choices, both of which we define later. Here, the notion of state $s$ can be generalized to any input data, such as an image for a visual classification task. In an active learning context, the agent may be able to select the state that maximizes the information gain over $\omega$ from its interaction, e.g. by selecting the most informative datapoint to be labeled [Kapoor et al., 2007] or changing the behavior of other agents in the environment [Sadigh et al., 2016]. Otherwise, the state remains static, and the agent’s objective is to select an action query that maximizes the information gain over $\omega$ within that state.

Alternatively, information gain can be expressed as the expected Kullback–Leibler divergence of the prior distribution from the posterior belief distribution over model weights:

$$\Phi_i(\omega, s) = \max_{q \in Q_i(\omega, s)} \mathbb{E}_{c \in C_i(q)}[D_{KL}(p(\omega|c)||p(\omega))]$$

Both formulations introduce three key, interaction-specific functions: $Q_i(\omega, s)$, $C_i(q)$, and $H(\omega|c)$ (used in Eqn. 2) or $D_{KL}(p(\omega|c)||p(\omega))$ (used in Eqn. 3). We describe these functions and their relationship with interaction types in the reminder of this section.

4.1 Query: $Q_i(\omega, s)$
A query $q$ is a specific set of data that an agent requests feedback on during a single instance of an interaction. $Q_i(\omega, s)$ is then the set of all possibly queries that can be posed to the teacher, given $\omega$ and $s$. In a showing interaction, the agent queries the teacher for an example action, or series of actions (trajectory). Therefore, there is only one possible query in the set $Q_i(\omega, s)$: the agent requests a demonstration from state $s$ without providing any additional information to the teacher. In a sorting interaction, the agent’s query consists of some $n$ trajectories originating from state $s$ (e.g., the teacher might be asked to order $n$ trajectories with respect to their effectiveness). If we assume that there are $k$ feasible trajectories originating from state $s$, then $|Q_i(\omega, s)| = \binom{k}{n}$. In both categorizing and evaluating interactions, which differ on the basis of their choice space and choice implications, an agent queries the teacher for feedback on a proposed trajectory, and so $|Q_i(\omega, s)| = k$.

4.2 Choice Space: $C_i(q)$
Once a query has been selected, the process for expanding a query into a set of possible explicit and implicit choices available to the teacher is also interaction-specific [Jeon et al., 2020; Koppol et al., 2021]. For example, both a categorizing and an evaluating interaction consist of querying the teacher by proposing a series of actions (e.g. a motion trajectory for a robot arm, or proposed labels for a set of object images). However, the set of feedback choices available to the teacher in response to an individual query varies by interaction. In the categorizing interaction, the teacher may be presented with a set of $\pm 1$ rating choices over the agent’s entire proposed sequence of actions. In the evaluating interaction, however, the teacher may observe the same sequence of actions but provide feedback at a finer scale, such as $\pm 1$ ratings on segments of the agent’s manipulation trajectory rather than a single rating over the full trajectory. An alternative evaluating interaction may involve providing corrections instead of critiques, enabling the teacher to interrupt the agent’s actions in real-time to provide alternative actions. That is, the teacher
must choose whether to interrupt the agent’s action at each time step, after which they must also choose what alternative action the agent should take. Thus, teachers make more feedback choices in response to a single evaluation query than a single categorizing query.

Overall, these examples illustrate the effect of interaction type on the choice set available to the teacher. These effects are apparent both across different interaction types (e.g., the density of the feedback resulting from a categorizing interaction versus an evaluating interaction), as well as within the same interaction type (e.g., critiques and corrections are both evaluating-type interactions, but result in different feedback choices that are available to the teacher).

Furthermore, the likelihood of the choice set containing the optimal choice is dependent on the quantity and quality of that set. For interaction types that provide an infinite set of query responses, such as a showing interaction, the teacher may provide feedback from an infinite set of options. In evaluating interactions, the teacher is also provided an additional option of whether to provide feedback or not. As a result of the infinite quantity of choices, the optimal choice must be contained within this set of options.

For interaction types that provide a finite set of query responses, such as sorting interactions, the quantity of choices available to the teacher are limited, and so the training data is dependent on the quality of the choices presented to the teacher. The quality of a choice set may be defined by the informativeness of each possible choice, estimated through an information gain formulation [Biyik et al., 2020].

4.3 Choice Implications: $H(\omega | c)$ or $D_{KL}(p(\omega|c)||p(\omega))$

The implications of the teacher’s choice on the agent’s training data is also dependent on the interaction type. In an information gain context, this implication can be represented as the conditional entropy over the model’s parameters given the feedback that the teacher did and did not provide [Jeon et al., 2020]. When learning from showing interactions, such as demonstrations, existing work in inverse reinforcement learning typically assumes that the teacher’s feedback represents the optimal action that the agent should take and updates the agent’s reward model accordingly [Abbeel and Ng, 2004; Ramachandran and Amir, 2007]. The demonstrations may also be used to learn a nonlinear cost function that represents the dynamics of the demonstrated task [Finn et al., 2016].

In categorizing interactions, the teacher’s feedback may be used to directly learn a regression model of the reward function that replicates their feedback (as shown by the TAMER framework [Knox and Stone, 2009; Warnell et al., 2018]). By training an action model separately from the reward model, improvements in the action model may be used to guide the agent’s queries to improve its reward model [Daniel et al., 2014]. However, feedback does not always reflect the reward of the agent’s state. Thomaz et al. [2006] show how categorizing feedback not only reflects the teacher’s feedback on the agent’s prior actions, but also feedback on their expectations of the agent’s future behavior. As a result, a key challenge is determining which states and/or state features correspond to the teacher’s feedback [Knox and Stone, 2009]. Furthermore, this feedback may correspond more closely to an “advantage function” that reflects the advantage of choosing a particular action in the agent’s current state, rather than the reward of entering the state itself [Mnih et al., 2016].

In sorting interactions, the training implications of the teacher’s choice is dependent on the other choices available to them. A pairwise preference between two actions may be interpreted as a loss function representing the margin between the agent’s predicted preference over the two options (according to its reward function) and the human’s actual preferences [Christiano et al., 2017]. As a result, the objective of the model is not necessarily to estimate an action’s reward itself, but rather, to learn a reward function that preserves the relative ranking of one action over another [Liu et al., 2017]. Learning from relative rankings has an added benefit: by removing the assumption that either of the ranked options is optimal, the model can learn a reward function that exceeds the performance of the teacher [Brown et al., 2019]. Since the strength of the teacher’s preference is unknown, it may be beneficial to provide an option to indicate equal preference between two options rather than force the teacher to indicate a preference [Holladay et al., 2016].

In evaluating interactions, the teacher provides feedback over a series of proposed or executed actions by the agent. This feedback must be considered with respect to the actions before and after the teacher’s feedback. For example, Celemim and Ruiz-del Solar [2019] presents a method for approximating the magnitude of the teacher’s binary feedback based on the variability of that feedback over time. In corrective interactions, the teacher’s feedback can be interpreted as an alternate demonstration that results in higher reward or performance than the agent’s originally proposed action. This correction can be used to update the agent’s behavior in real-time [Bajcsy et al., 2017] or interpreted as a singular sample of the desired change in the agent’s model (reinforced through additional corrections) [Fitzgerald et al., 2019].

Implicit and Explicit Information. While we have considered the training implications of the teacher’s explicit responses to the agent’s queries, the teacher may provide additional data that may be incorporated into the agent’s training process. For example, they may also reveal additional, implicit information via gestures [Breazeal et al., 2005], facial expressions [Cui et al., 2020], gaze [Zhang et al., 2020], or other social cues. The teacher’s lack of feedback in some states may also provide implicit data, such as when ignoring a web link or skipping a video [Bayer et al., 2017]. Interpreting a teacher’s silence as positive feedback may speed up learning [Cederborg et al., 2015]; however, the direction and magnitude of reward implied by a teacher’s silence is likely to be domain-specific. Furthermore, higher-level information about the task may be learned implicitly through multiple interactions [Niekum et al., 2015]. Leveraging both implicit and explicit information may result in increased informativeness of an interaction without asking of any additional effort from the teacher.
5 User Experience and Teaching Quality

In Section 2, we discussed how, in a HIL-ML setting, “human performance” on a task equates to their teaching performance. We now survey how interaction types influence human performance, discuss metrics of the effects of interaction types on human performance, and review how human performance can influence training data.

5.1 Interactions Affect Human Performance

Different interaction types have been shown to differently influence human factors such as ease of use, cognitive load [Koppol et al., 2021], and perception of the learner [Cakmak and Thomaz, 2012]. However, no existing work has directly studied the effects of these human factor differences on learning outcomes. We can consider these human factors as facets of a human teacher’s mental model of an interaction, which includes the teacher’s model of the learning agent (e.g. capability, performance) and their own perceived task as a teacher. We subsequently summarize a few ways in which such a mental model may affect human performance.

Passive vs Active Learner. Passive learning involves the agent using a training set that is defined irrespective of its learning status. Active learning enables the learner to query for informative data points, thus improving its sample efficiency [Settles, 2012]. The relationship between the learner and the user varies dramatically depending on whether learning is passive or active [Cakmak et al., 2010] due to the co-adaptive nature of active learning [Dudley and Kristensson, 2018], and also varies due to the type of queries posed during active learning [Chao et al., 2010].

Offline vs Online Learner. The learning agent interacting and querying the human teacher may update its model during an interaction, and can update at different frequencies, ranging from fully offline to fully online. An online learner will collect a batch of data, update its model, and demonstrate its improved performance before receiving additional feedback. This provides the teacher with feedback on how well the agent is learning, and allows them to change their teaching strategy by providing targeted data according to the model state [Kronander and Billard, 2012; Kulesza et al., 2015]. In contrast, an offline learner collects a single training set and performs no model update during the interaction, thus requiring less time and interaction effort from the teacher.

Pedagogical vs Pragmatic Teaching. People have been shown to demonstrate tasks differently if they know that a learner is attempting to learn from them, as opposed to if they are asked to complete it as efficiently as possible [Fisac et al., 2020]. Pedagogical human teachers may intentionally take sub-optimal actions in order to communicate more information in a single query response [Ho et al., 2016].

Mental Model of the Agent’s Learning Status. In addition to having a mental model of their interaction with the agent, the teacher may also have a mental model of the agent’s learning status as well; that is, the agent’s current knowledge and performance over the task. A teacher’s mental model can affect various factors of their task performance including planning, persistence, and satisfaction [Jih and Reeves, 1992]. As a result, it is important to consider how this mental model may be affected by the agent’s performance and the interaction between the teacher and agent. Hedlund et al. [2021] found that agent performance can affect a teacher’s mental model of both the agent and their own teaching capability. Krening and Feigh [2018] showed how teachers perceived an agent trained using verbal demonstrations as being more intelligent and better-performing than the one trained through binary critiques. Furthermore, interactive learning methods lent themselves to more accurate assessments of agent capability as compared to passive, supervised learning [Cakmak et al., 2010].

5.2 Indicators of Human Performance

While “ground-truth” for optimal human performance in HIL-ML systems may not exist, there are measures that are known to affect human performance. In particular, an increase in workload has been correlated with a decrease in task performance [Sweller, 1988; Prewett et al., 2010]. Workload can be measured both in subjective, self-reported measures and in objective task measures [Longo, 2018]. The NASA-TLX survey [Hart and Staveland, 1988] has been widely adopted in human factors research to measure subjective workload, and consists of several sub-metrics including mental demand, physical demand, temporal demand, performance, effort and frustration. The popular System Usability Scale (SUS) is a validated survey that provides a subjective measure of the usability of any given system, reflecting measures such as users’ ease and confidence when using a system. A strong, positive association exists between task performance and subjective satisfaction with an interface [Nielsen and Levy, 1994]. Workload and usability have also been found to be non-overlapping measures in an HCI task, which suggests that combining them may provide a more accurate prediction of objective task performance [Longo, 2018].

5.3 Direct Effects of Human Performance on Training Data

In Section 3, we described how noisiness in the data, as well as the quantity and distribution of collected data affect learning outcomes. We now address how human performance can directly influence those factors.

Noise. We focus on noise introduced via human error (e.g., where a human teacher fails to provide the conventional ground truth). For example, data collected from crowdworkers can be low quality, as workers are incentivized to maximize their own earnings at the potential expense of providing thoughtful labels [Hsueh et al., 2009]. Noisiness can also arise from human teachers without adversarial intentions, due to factors such as the amount of precision afforded by a particular user interface [Aker et al., 2012].

Distribution. Collecting well-distributed data that captures domain shifts is critical for robust models. The availability of crowdworkers suggests the possibility of increased diversity in dataset curation [Lease, 2011], and has already been shown
to manifest in more efficient exploration and learning [Mandlekar et al., 2018]. The teaching interaction may be adapted in response to poorly-distributed training data; for example, tasking a teacher with finding a positive example in an underrepresented region of the state space [Lin et al., 2018]. However, interaction mechanisms that are not designed to be accessible and usable by a variety of individuals may result in datasets that either eschew or result in low-accuracy feedback from entire swaths of people [Vashistha et al., 2018].

Quantity. In HIL-ML systems, the interaction type being leveraged can affect the rate of data collection, and ultimately limit the amount of data collected. Demonstrations on physical robots, for example, can be difficult and time-consuming to provide; simulated approaches with user-tested interfaces can increase labeling throughput and lead to better learned policies [Mandlekar et al., 2018; Kent et al., 2017].

6 Challenges and Open Questions

Benchmarks have played an important role in modern advances of machine learning through facilitating standard datasets and environments for fair comparisons. Existing HIL-ML benchmarks exist in the form of datasets or use expert models obtained from training reinforcement learning algorithms, bypassing active interactions with human teachers, and therefore do not provide a mechanism for comparing across different interaction types. Developing novel methods to efficiently evaluate HIL-ML systems with real/simulated human inputs will be beneficial to the research community.

To systematically compare the effects of different interaction types, it is important to have a standard measure for teaching cost that applies to all of them. Rigter et al. [2020] demonstrate how a robot may moderate its own autonomy in a shared autonomy setting in order to minimize both interaction and failure cost, but assumes a hand-coded measure of both cost values. Teaching cost of an interactive learning algorithm has been measured primarily as the interaction duration and/or subjective cognitive load [Racca et al., 2019; Cui et al., 2019; Jauhri et al., 2020]. However, interaction duration alone does not capture all the aspects of teaching cost, and subjective measures of cognitive load tend to have huge variance across people. Recent work of Bryk et al. [2020] proposes to measure teaching cost of a single comparison query as a function of interaction duration, complexity of the question, and similarity to past queries. A unified set of metrics for evaluating teaching cost across different interaction types is needed.

Modeling human behaviors in HIL-ML systems is not only important for interpreting collected data but also crucial for analyzing the effects of different interaction types on learning outcomes. Despite rich evidence in psychology research that humans are not rational decision makers [Arkes and Ayton, 1999; Hewig et al., 2011], many existing methods have been assuming rational or noisily rational human teachers and the same human teacher model has been employed by various algorithms that learn from different interaction types [Sadigh et al., 2017; Jeon et al., 2020]. With different systematic biases known to exist in human decision making [Shah et al., 2019] and known behavior differences under different settings (such as pedagogical vs pragmatic), it is important to understand how these factors interact with the design of interaction types. At the same time, crowdsourcing has become a promising way of acquiring large-scale human annotated data [Vaughan, 2017; Osentoski et al., 2010]. Designing learning systems that will interact with many different users and collect data from them brings additional challenges for modeling teaching behaviors.

Given the complex relationship between interaction types and training data, there may not be a single best interaction type to use for a particular task. The optimal solution may arise from combining different types of interaction types. Work of Ibarz et al. [2018] and Palan et al. [2019] leverage multiple types of interaction. Bullard et al. [2018] arbitrates between showing and categorizing. The work of Jeon et al. [2020] proposes a way to optimize for interaction types for reward learning from the information gain perspective but does not take human performance into account.

Understanding how social biases [Fiske, 2016] can be introduced during data collection has been identified as an increasingly important component of building fair and responsible ML models [Liu et al., 2018; Drozdowski et al., 2020]. Our proposed relationship graph identifies two pathways by which interaction types can influence training data, and may provide a new perspective on sources of bias in HIL-ML system development.

In summary, we have surveyed existing literature on interaction types, established a relationship graph outlining and justifying effects of interaction types on learning outcomes, and presented a unifying representation of the training data implications of these interaction types. We anticipate that this comprehensive overview of the role of interaction types in HIL-ML systems will support future research that leverages, compares, or constructs interactions for learning.

Acknowledgements

The authors would like to thank Reuben Aronson, Stephen Giguere, W. Bradley Knox, Mike Lee, Michelle Zhao, and the anonymous reviewers for their valuable feedback. This work was conducted at CMU and UT Austin, and was supported in part by the Office of Naval Research (N00014-18-1-2503, N00014-18-2243), the National Science Foundation (IIS-1724157, IIS-1638107, IIS-1749204, IIS-1925082), AFOSR (FA9550-20-1-0077), and ARO (78372-CS).

References


