Teaching Humans When To Defer to a Classifier via Exemplars

Hussein Mozannar, Arvind Satyanarayan, David Sontag

Massachusetts Institute of Technology
mozannar@mit.edu

Abstract

Expert decision makers are starting to rely on data-driven automated agents to assist them with various tasks. For this collaboration to perform properly, the human decision maker must have a mental model of when and when not to rely on the agent. In this work, we aim to ensure that human decision makers learn a valid mental model of the agent's strengths and weaknesses. To accomplish this goal, we propose an exemplar-based teaching strategy where humans solve a set of selected examples and with our help generalize from them to the domain. We present a novel parameterization of the human's mental model of the AI that applies a nearest neighbor rule in local regions surrounding the teaching examples. Using this model, we derive a near-optimal strategy for selecting a representative teaching set.We validate the benefits of our teaching strategy on a multi-hop question answering task with an interpretable AI model using crowd workers. We find that when workers draw the right lessons from the teaching stage, their task performance improves. We furthermore validate our method on a set of synthetic experiments.

Introduction

Automated agents powered by machine learning are augmenting the capabilities of human decision makers in settings such as healthcare (Gaube et al. 2021), content moderation (Link, Hellingrath, and Ling 2016) and more routine decisions such as asking AI-enabled virtual assistants for recommendations (Shaikh and Cruz 2019). This mode of interaction whereby the automated agent serves only to provide a recommendation to the human decision maker, a setting typically named *AI assisted decision making*, is the focus of our study here. A key question is how does the human expert know when to rely on the AI for advice. In this work, we make the case for the need to initially onboard the human decision maker on when and when not to rely on the automated agent. This allows the human to have an accurate mental model of the AI agent, and this mental model helps in setting expectations about the performance of the AI on different examples.

Our onboarding phase consists of letting the human predict on a series of specially selected teaching examples while providing feedback and enabling lesson retention through letting the human write down rules indicating what they learned from each example. Our approach is inspired by research in the education literature that highlight the importance of feedback and lesson retention for learning (Atkinson et al. 2000; Hattie and Timperley 2007). To select the teaching examples, we need to have a mathematical framework of how the human mental model evolves after we give them feedback. We model the human thought process as first deciding whether to rely on the AI's prediction or not using an internal *rejector*. This rejector is what we refer to as the human's mental model of the AI. We propose to model the human's rejector as consisting of a prior rejector and a nearest neighbor rule that only applies in local regions surrounding each teaching example. This novel parameterization is inspired by work in cognitive science that suggests that humans make decisions by weighing similar past experiences (Bornstein et al. 2017). Assuming this rejector model, we give a near-optimal greedy strategy for selecting a set of representative teaching examples that allows us to control the examples and the region surrounding them.

We first evaluate the efficacy of our algorithmic approach on a set of synthetic experiments and its robustness to the misspecification of the human's model. For our main evaluation, we conduct experiments on Amazon Mechanical Turk on the task of passage-based question answering from HotpotQA (Yang et al. 2018). Crowdworkers first performed a teaching phase and were then tested on a randomly chosen subset of examples. Our results demonstrate the importance of teaching: around half of the participants who undertook the teaching phase were able to correctly determine the AI's region of error and had a resulting improved performance. The full version of this paper is available on arxiv (Mozannar, Satyanarayan, and Sontag 2021).

Related Work

One of the goals of explainable machine learning is to enable humans to better evaluate the correctness of the AI's prediction by providing supporting evidence (Lai and Tan 2019; Suresh et al. 2021; Wortman Vaughan and Wallach 2021). However, these explanations do not inform the decision maker how to weigh their own predictions against those of the AI or how to combine the AI's evidence to make their final decision (Kaur et al. 2020). The AI explanations cannot factor in the effect of the human's side information, and thus the human has to learn what their side information reveals about the performance of the AI or themselves. Moreover, if the AI's explanations are unfaithful or become so due to a distribution shift in the data (DeVries and Taylor 2018), then the human may then over-weigh the AI's abilities. Another direct approach for teaching is presenting the human with a set of guidelines of when to rely on the AI (Amershi et al. 2019). However, these guidelines need to be developed by a set of domain experts and no standard approach currently exists for creating such guidelines. As a byproduct of our teaching approach, each human writes a set of unorganized rules that can then be more easily turned into such guidelines.

The reverse setting, of teaching a classifier when to defer to a human, is dubbed as learning to defer (LTD) (Madras, Pitassi, and Zemel 2018; Raghu et al. 2019; Mozannar and Sontag 2020; Wilder, Horvitz, and Kamar 2020). The main goal of LTD is to learn a rejector that determines which of the AI and the human should predict on each example. However, there are numerous legal and accountability constraints that may prohibit a machine from making final decisions in high stakes scenarios. Additionally, the actual test-time setting may differ from that which was used during training, but since in our setting the human makes the final decision, this allows them to adapt their decision making and detect any unexpected model errors. As an example in a clinical use case, factors such as times of substantially increased patient load may affect the human expert's accuracy. The human may also occasionally have side-information that was unavailable to the AI that could improve their decision making. Compared to LTD, deployment may be simplified because the same AI is used for all experts; as new experts arrive, our onboarding phase trains them to use the AI according to their unique abilities. Our teaching setting and LTD also use very different techniques. Although the objective that we present in Equation (2) is closely related to the objective used by Mozannar and Sontag (2020), the main task in our setting is that of teaching the human when to defer. This requires us to develop a formalization of the human mental model and algorithms for selecting a subset of examples that enables accurate learning.

Related work has explored how to best onboard a human to trust or replicate a model's prediction. LIME, a black-box feature importance method, was used to select examples so that crowdworkers could evaluate which of two models would perform better (Ribeiro, Singh, and Guestrin 2016; Lai, Liu, and Tan 2020). Their selection strategy does not take into account the human predictor, nor does their approach do more than display the examples which is what we contribute. On a task of visual question answering, Chandrasekaran et al. (2018) handpicked 7 examples to teach crowdworkers about the AI abilities and found that teaching improved the ability to detect the AI's failure. Feng and Boyd-Graber (2019) on a Quizbowl question answering task highlight the importance of modeling the skill level of the human expert when designing the explanations. Through a study of 21 pathologists, Cai et al. (2019) gathered a set of guidelines of what clinicians wanted to know about an AI prior to interacting with it. Bansal et al. (2019) investigate the role of the human's mental model of the AI on task accuracy, however, the mental model is formed through test time interaction rather than through an onboarding stage. Bansal et al. (2021) propose a theoretical model for AI-assisted decision making, assuming that the human has a perfect mental model of the AI and that the human has uniform error. Finally, our work was inspired by the literature on machine teaching (Singla et al. 2014; Hunziker et al. 2018) and curriculum learning (Graves et al. 2017).

Problem Setup

Our formalization is based on the interaction between two agents, the AI, an automated agent, and a human expert who both collaborate to predict a target $Y \in \mathcal{Y}$ based on a given input context. The AI consists of a predictor $\pi_Y : \mathcal{X} \to \mathcal{Y}$ that can solve the task on its own and a policy $\pi : \mathcal{X} \to \mathcal{A}$ which serves to communicate with the human and sends them a message A. The message space $\mathcal{A}$ may consist for example of the AI's prediction $\pi_Y(X)$ alongside an explanation or a confidence score for their decision. The human expert then integrates the AI message A and their view of the input $Z \in \mathcal{Z}$ to make a final decision $M(Z, A)$ which can either be to predict on their own or allow the AI to predict. The input space of the human and AI X and Z could be different since the human may have side information that the AI can't observe. The human consists of a **predictor** $h : \mathcal{Z} \times \mathcal{A} \to \mathcal{Y}$ parameterized by θ_h and the human decides to allow the AI to predict or not according to a **rejector** $r : \mathcal{Z} \times \mathcal{A} \to \{0, 1\}$ parameterized by θ_r, where if $r(Z, A; \theta_r) = 1$ the human uses the AI's answer for its final prediction. This implies that the final human decision M is as follows:

$$M(Z, A) = \begin{cases} \pi_Y(x) & \text{, if } r(Z, A; \theta_r) = 1 \\ h(Z, A; \theta_h) & \text{, otherwise} \end{cases} \quad (1)$$

System objective. Given the above ingredients and a performance measure on the label space $l(y, \hat{y}) : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+$ (e.g. 0-1 loss), the loss that we incur is the following:

$$\begin{aligned} L(\pi, \pi_Y, h, r) =& \mathbb{E}_{x,z,y}[l(\pi_Y(x), y)\mathbb{I}_{r(x,\pi(x))=1} + \\ & l(h(z, \pi(x)), y)\mathbb{I}_{r(x,\pi(x))=0}] \end{aligned} \quad (2)$$

The central Human-AI interaction problem. Given a fixed AI policy, and human parameters (θ_h, θ_r), the manner in which the human expert integrates the AI's message depends only on the expert context Z and the message itself A. It is more realistic to assume that the expert has a *mental model* of the policy π that they have arrived at from either a description of the policy or from previously interacting with it; the rejector here formalizes the *mental model*. This insight forces us to now consider the parameters (θ_h, θ_r) as variables that are learned by the human as a function of the underlying AI policy π. This makes the optimization of the loss now much more challenging as whenever the policy π changes, the human's mental model, (θ_h, θ_r), needs to update. Therefore, we need to first understand how the human's mental model evolves and how we can influence it.

Teaching Humans about the AI. In this work, we focus on exemplar based strategies to allow the human to update their mental models of the AI. The question is then how do we select a minimal set of examples that teaches the human an accurate mental model of the AI. To make progress, we need to first understand the form of the human's rejector and how it evolves, which we elaborate on in the following section. Crucially, we will keep the AI in this work as a fixed policy and not look to optimize for it. Once we understand this first step, future work can then look to close the loop.

Human Mental Model

We now introduce our model of the human's rejector and the elements of the teaching setup. The tasks we are interested in are where humans are *domain experts*, meaning that their knowledge about the task and their predictive performance are fixed. We further extend this to how they may incorporate the AI message in their prediction, but crucially not how they decide when to use the AI. This assumption translates in our formulation as follows.

Assumption 1 *The human predictor does not vary as they interact with the AI, i.e. we assume θ_h to be fixed.*

We now move our attention to the human's rejector, which represents their mental model of the AI, and learned after observing a series of labeled examples. Research on human learning from the cognitive science literature has postulated that for complex tasks humans make decisions by sampling similar experiences from memory (Bornstein et al. 2017; Giguère and Love 2013; Richler and Palmeri 2014). Moreover, (Bornstein et al. 2017) makes the explicit comparison with nearest neighbor models found in machine learning. However, standard nearest neighbor models don't allow for prior knowledge to be incorporated. For this reason, we postulate a nearest neighbor model for the human rejector that starts with a prior and updates in local regions of each shown example in the following assumption.

Assumption 2 (Form of Human's rejector) *The human's rejector consists of a prior rejector rule and a nearest neighbor rule learned after observing teaching examples $D_T = \{z_i, a_i, r_i\}_{i=1}^m$.*

Formally, let $g_0(Z, A) : \mathcal{Z} \times \mathcal{A} \to \{0, 1\}$ be the human's prior rejector. Figure 1 illustrates the scenario: the prior is the region at the boundary of the human predictor h. Let $K(.,.) : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}^+$ be the similarity measure that the human employs to measure the degree of similarity between two instances. The human's rejector uses a learned rule if they had observed an example similar with respect to $K(.,.)$ during teaching, otherwise falling back on their prior:

$$r(Z, A; \theta_r) = \begin{cases} vote(B(Z)) & , \textit{if } B(Z) \neq \emptyset \\ g_0(Z, A) & , \textit{otherwise} \end{cases} \quad (3)$$

where $B(Z)$ is the set of all points in D_T that they observed in training sufficiently similar to Z: $B(Z) = \{i \in [m] \mid K(Z, z_i) > \gamma_i\}$. The degree of similarity is measured by a scalar γ_i that the human sets for each teaching example, in figure 1 all the points in the shaded ball have $B(Z) = \{z_1\}$. The rule $vote(B(Z))$ defines the label for all points similar to Z based on a weighted decision: $vote(B(Z)) =$ $\arg\max_{k \in \{0,1\}} \frac{\sum_{i \in B(Z)} \mathbb{I}\{r_i = k\} K(Z, z_i)}{\sum_{i \in B(Z)} K(Z, z_i)}$ *Where r_i is the deferral rule that the human has learned on example z_i.*

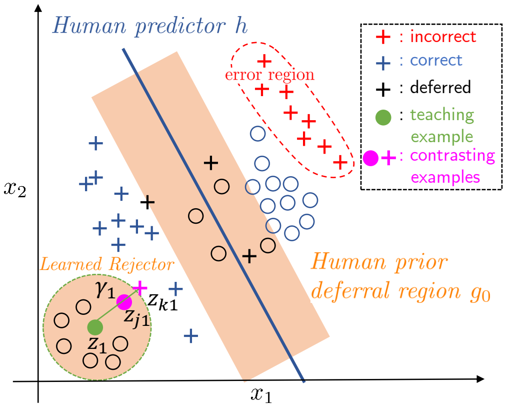


Figure 1: The task is classification with labels $\{o, +\}$, the human prediction h is the blue line and the prior g_0 is the shaded orange region surrounding the boundary. Points in red is where the human is incorrect, in blue correct and in black point deferred to the AI. The AI is assumed to be correct on examples far from the human boundary. The human receives a teaching example z_1 (in green) with radius γ_1. Also shown are the two contrasting examples z_{j1} and z_{jk} (in pink) that define the region.

Discussion on the Assumptions. In our assumptions above, we assumed knowledge of the following parameters: the human predictor $h(Z, A)$, the prior human rejector $g_0(Z, A)$ and the human similarity measure $K(,,.)$. In fact, as we will see, we only need to know the expert error distribution $\mathbb{E}[l(h(Z, A), Y)|Z, A]$ rather than the full expert predictor; it may be reasonable to estimate the expert's error distribution from previously collected data. The prior rejector g_0 can also be learned by testing the human prior as evidenced by prior work on capturing human priors (Kim et al. 2019; Bourgin et al. 2019), otherwise, a reasonable guess is the human deferring by just thresholding their own error rate. Finally to teach the human, we need a proxy for the similarity measure $K(.,.)$. This can be obtained in many ways: one can learn this metric with separate interactions with the human, see (Ilvento 2019; Qi et al. 2009), or rely on an AI based similarity measure e.g. from neural network embeddings (Reimers and Gurevych 2019). This last proxy is readily available and in the framework of our study, we believe it is reasonable to use. An important part of the rejector is the associated radius γ_i with each teaching example i, the radius allows the human to generalize from each teaching example to the entire domain. The human learning process leaves the setting of γ_i completely up to the human and is not observed. However, we hope to directly influence the value of γ_i that the human sets during teaching.

Teaching a Student Learner

Formulation. The previous section introduced the model of the human learner, in this section we will set out our approach to select the teaching examples for the onboarding stage. We assume access to a labeled dataset $S =$

$\{x_i, z_i, y_i\}_{i=1}^n$ that is independent from the training data of the AI model. For each point we can assign a deferral decision r_i that the human should undertake that minimizes the system loss. Explicitly, the optimal deferral decision r_i is defined to select who between the human and AI has lower loss on example i: $r_i = \mathbb{I}\{\mathbb{E}[l(h(z_i, a_i), y_i)] \geq \mathbb{E}[l(\pi_Y(x_i), y_i]]\}$. Note that to derive r_i we only need to know the loss of the human on the teaching set and not their predictions. Define then $S^* = \{x_i, z_i, r_i\}_{i=1}^n$ as a set of examples alongside deferral decisions. As mentioned previously, the human is also learning a radius γ_i with each example. The radius γ_i should be set large enough to enable generalization to the domain, but small enough for the region to be coherent so that the human can interpret why should they follow the optimal deferral decision.

Let $D_z \subset S^*$ and let D_γ be the set of radiuses associated with each point in D_z and define $D = (D_z, D_\gamma)$. Define the loss of the human learner $M(.,.; D)$ now only parameterized by the teaching set D as follows:

$$L(D) = \sum_{i \in S} l\left(M(z_i, a_i; D), y_i\right) \tag{4}$$

Greedy Selection. Note that since the radiuses set by the human are learned only after observing the example, we try to jointly optimize for the teaching point and the radius to teach. To optimize for D, consider the following greedy algorithm (GREEDY-SELECT) which starts with an empty set D_0 , and then repeats the following step for $t = 1, \cdots, m$ to select the example z and radius γ that leads to the biggest reduction of loss if added to the teaching set:

$$z, \gamma = \arg \min_{z_i \in S \setminus D_t, \gamma} L(D_t \cup \{z_i, \gamma\}), \tag{5}$$

$$\text{s.t. } \exists k \in [n] \; s.t. \; \gamma = K(z_i, z_k), \tag{6}$$

$$\text{and } \frac{\sum_{j \in [n], K(z_i, z_j) > \gamma} \mathbb{I}_{r_j = r_i}}{|\{j \in [n], K(z_i, z_j) > \gamma\}|} \geq \alpha \tag{7}$$

Constraint (6) restricts γ to be the similarity between z and another data point and constraint (7) ensures that $\alpha\%$ of all points inside the ball centered at z share the same deferral decision as z. The scalar α is a hyperparameter that controls the consistency of the local region: when $\alpha = 1$, the region is perfectly consistent and we call this setting CONSISTENT-RADIUS, and when $\alpha = 0$ the constraint is void and we dub the algorithm as DOUBLE-GREEDY. Note that the radius γ is actually defined by two points: the point z_k in equation (6) that defines the boundary and an interior point z_j that is the least similar point to z with similarity at least γ; these two points are illustrated in Figure 1 with the color pink. These two points must actually share opposing deferral actions with $r_k \neq r_j$ and thus are contrasting points later used as a way to describe the local region.

Theoretical Guarantees. Let D_t be the solution found by the greedy algorithm and D^* the optimal solution. We now try to see how we can compare D_t to D^*. We restrict our attention to the case of $\alpha = 1$; when $\alpha < 1$ the guarantees may not hold. We can derive a guarantee on the gap of performance of our algorithm versus the optimal teaching set as the next theorem demonstrates.

Theorem 1 *Let $F(X) = L(\emptyset) - L(X)$, when $\alpha = 1$, $F(.)$ is submodular, monotone and positive. Moreover, the GREEDY-SELECT algorithm described above achieves the following performance compared to the optimal set D^*:*

$$\underbrace{L(D_m)}_{\textit{loss of chosen set}} \leq (1 - \frac{1}{e}) \underbrace{L(D^*)}_{\textit{loss of optimal set}} + \frac{1}{e} \underbrace{L(\emptyset)}_{\textit{loss of prior rejector}}$$

The proof can be found in the appendix. Theorem 1 gives a guarantee on the subset chosen by the greedy algorithm with an $1 - \frac{1}{e}$ approximation factor, one can ask if we can do better. We prove that a generalization of our problem is in fact NP-hard in the appendix.

Human Teaching Approach. After running our greedy algorithm, we obtain a teaching set D that we now need to teach to the human. We rely on a four stage approach for teaching on each example so that they are able to learn and generalize to the neighborhood around it. The human first predicts on the example z, then they receive feedback on their prediction and the AI's prediction. We then show them a description of the region around the example that helps them learn the radius. Specifically, we show them the two contrasting examples z_j and z_k defined by γ_i and high level features about the neighborhood. Finally, we ask them to formalize in writing a rule describing the region and the action to take inside that region. This rule that they write per example helps the human in creating a set of guidelines to remember for when to rely on the AI and ensures that they reflect on the teaching material.

Experimental User Study

Experimental Task and Dataset. Our focus will be on *passage-based question answering* tasks. These are akin to numerous real world applications such as customer service, virtual assistants and information retrieval. It is of interest as relying on an AI can reduce the time one needs to answer questions by not reading the entire passage and as an experimental setup it allows a greater range in the type of *sub-expertise* we can allow for compared to experimental tasks in the literature. We rely on the HotpotQA dataset (Yang et al. 2018) collected by crowdsourcing based on Wikipedia articles. We slightly modify the HotpotQA examples for our experiment by removing at random a supporting sentence from the two paragraphs. The supporting sentence removed does not contain the answer, so that each question always has an answer in the passage, however, it may not always be possible to arrive at that answer. This was done to make the task harder and create incentives for expert humans to use the AI. We further remove yes/no questions from the dataset and only consider hard multi hop questions from the train set of 14631 examples and the dev set of 6947 examples.

Simulated AI. One of the top performing models on HotpotQA is SAE-large: a graph neural network on top of RoBERTa embeddings (Tu et al. 2020). We performed a detailed error analysis in the appendix of the SAE-large model predictions on the dev set. However, our analysis uncovered only few and small regions of model error. For our experimental study, we want to evaluate the effect of teaching in two ways: 1) through systematically checking the validity

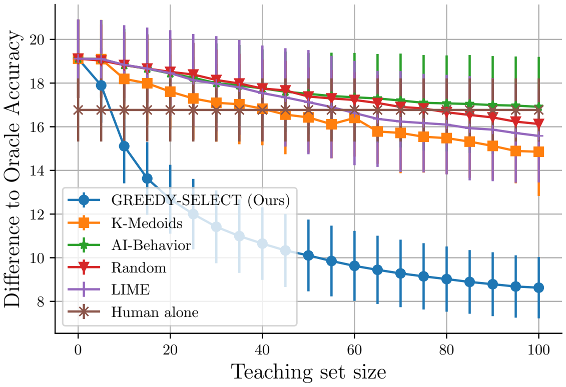


Figure 2: Teaching set size versus the negative difference between the human's learner test accuracy under the different methods compared to ORACLE. We plot for setting B where $(\alpha_{ai} = 1, \beta_{ai} = 1), (\alpha_h = 2, \beta_h = 1)$ and $\epsilon = 0.9$ with CONSISTENT-RADIUS; the error bars denote standard deviation across 10 trials.

of the user lessons and 2) through objective task metrics. The SAE model makes it harder for us to do both especially with a limited number of responses from crowdworkers. For this reason, we decided to create a simulated AI whose error regions are more interpretable. We first cluster the dataset using K-means with k_p clusters based on only the paragraph embeddings obtained from a pre-trained Sentence-BERT model (Reimers and Gurevych 2019). The simulated AI model is parameterized by a vector $err_p \in [0,1]^{k_p}$ where the probability of error of the AI on cluster i by $err_p[i]$.

Metrics. Our aim will be to measure objective task performance and effort through the proxy of time spent on average per example. Our task performance metric is the F1 score on the token level (Rajpurkar et al. 2016); we will measure this when considering the final predictions (Overall F1), on only when the human defers (Defer F1) and when the human does not defer (Non-Defer F1). We will also measure *AI-reliance*: this is calculated as how often they rely on the ”Let AI answer for you” button in Figure 3a.

Simulated Users

Before we experiment with real human users, we evaluate the teaching complexity, i.e. the relation between teaching set size and human accuracy, of our teaching algorithm on simulated human learners that follow our assumptions. We further evaluate the robustness of our approach when we do not have full knowledge of the human parameters.

AI and Human model. We use the simulated AI model with $k_p = 15$ and a vector of errors err_p where for each i, $err_p[i]$ is drawn *i.i.d.* from Beta$(\alpha_{ai}, \beta_{ai})$. The human predictor is analogous to the AI model with a different vector of probabilities err'_p sampled from Beta(α_h, β_h). The human prior thresholds the probability error of the human to a constant ϵ. Finally, the human similarity measure is the RBF kernel on the passage embeddings i.e. $K(x, x') = e^{-|x-x|^2}$. In this setup both the human and AI contexts are identical and the AI does not send any messages to the human.

Baselines. We implement a domain cover subset selection baseline in K-Medoids, the LIME selection strategy by (Ribeiro, Singh, and Guestrin 2016) with 10 features per example following (Lai, Liu, and Tan 2020) (LIME), random selection baseline (RANDOM) and a baseline that greedily selects the point that helps a 1-NN learner best predict the AI errors (AI-BEHAVIOR). Finally, we also compare to the optimal rejection rule computed with knowledge of the human and AI error rates by picking the lower one (ORACLE).

Condition	Oracle Gap @n=30
Full Information	6.38 ± 1.56
Missing g_0	6.90 ± 1.80
Noisy Radius	9.74 ± 3.0
Missing h	13.47 ± 5.07
No Information+Noise	15.12 ± 4.00
Prior only	16.72 ± 1.22
Human Alone	19.8 ± 2.80

Table 1: Test Accuracy gap between DOUBLE-GREEDY and ORACLE at teaching set of size 30 under various conditions. This is performed under setting B.

Experimental setup. We will compare to the baselines as we vary the size of the teaching set D_T. To illustrate the effectiveness of the teaching methods, we focus on two settings: A) the Human is less accurate than the AI but their prior rejector rarely defers where we set the following and B) the Human is more accurate than the AI but their prior rejector over defers to the AI. These two settings is where teaching is most beneficial as the prior is erroneous. We evaluate for each setting 10 different random settings of the human and AI error probability vectors and average the results.

Results. Figure 2 shows the gap between Oracle and human accuracy on the dev set compared to the size of the teaching set for each of the methods. We can see that our approach is able to outperform the baselines under setting B with CONSISTENT-RADIUS. We observe a wide gap between our method and the baselines, this is because the teaching examples here must focus on only a select number of the clusters and cover them sufficiently. In the appendix we show similar results for setting A and B with the DOUBLE-GREEDY strategy. With the greedy radius selection, we require fewer examples to reach high accuracy and the gap between our method and the baselines narrows.

Robustness to Misspecification of Human model. We evaluate accuracy when the human is not learning the correct radius; this simulates noise in the learning process. The radius γ_i that the human learns is a noisy version of $\hat{\gamma}_i$ where we add a uniformly distributed noise to it. We then evaluate when we have no knowledge of the prior rejector g_0 or/and no knowledge of the human predictor h. In our algorithm, we only need the predictions of g_0 and h on the teaching set, when we don't know either of these parameters, we replace them by a random binary vector Bernoulli$(1/2)^n$. Results are shown in Table 1. We can see that even if we don't have knowledge about the prior, accuracy is not impacted. However, if we don't have knowledge about the predictor h, then performance drops significantly. To evaluate how much information about h we need to teach the human, we learn a teaching set assuming the human's error probability has additive uniform noise: on setting B with DOUBLE-GREEDY, we can tolerate up to 0.25 error in knowledge about cluster error probability with no noticeable drop in performance. Note that when we don't have any knowledge about the human and the learning process is noisy, teaching is impacted.

Crowdsourced Experiments Details

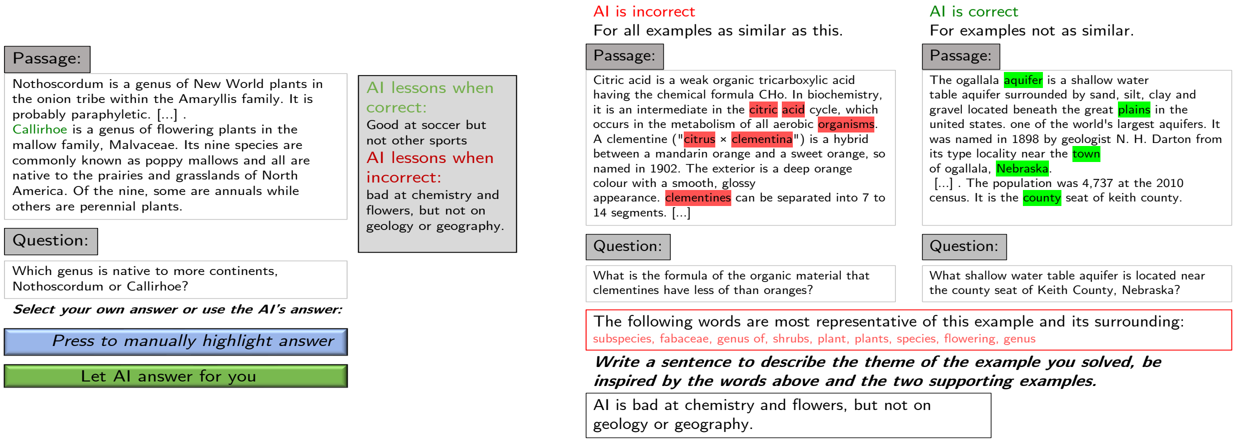
Testing user interface. Our user interface during testing is shown in Figure 3a which shows a paragraph and its associated question. The human can either submit their own answer or let the AI answer for them using a special button. However, the interface does not display the AI's answer or any explanation, which forces the user to rely solely on their mental model and the teaching examples to make a prediction. This was done so that we can control for the effect of teaching solely, as showing the AI prediction at test time leaks information about the AI beyond what was shown in the teaching set. Moreover, not showing the AI prediction forces the human to explicitly think about the AI performance. The right panel next to the passage shows the lessons that the user wrote down during teaching.

Teaching user interface. Following our teaching algorithm, during teaching, the worker is first faced with the same user interface as in test time. The difference is that *after* they answer, they receive feedback on the correctness of their answer and can see the AI's answer. We then show the human the two constrasting examples with LIME word highlights. As a high level description of the local region, we show the top 10 most weighted words obtained by LIME in the ball surrounding the original teaching example (Ribeiro, Singh, and Guestrin 2016) (see Figure 3b). After they observe the two supporting examples, they are asked to write a sentence that describes the lesson of the example. These sentences are available during test-time for the workers to review as help for answering new questions.

Experimental Design and Baselines. The experimental teaching setup proceeds in three stages. The first stage (Stage 0) is a tutorial that introduces the task with two examples. Stage 1 is the teaching stage where the worker solves 9 teaching examples and stage 2 is the testing phase where the worker solves 15 questions with no feedback.We randomly assign each participant to one of three conditions. In the first condition the participants go through the entire pipeline described above (Ours Teaching). The second is condition is called (LIME-Teaching) where LIME is first used to obtain 18 examples. During teaching, users are asked to solve the first 9 questions and are then shown: LIME highlights of the example, performance feedback and asked to write a lesson of what they learned. Then users view the 9 remaining examples with LIME highlights without needing to solve them or write lessons. The difference with our method is that workers don't see the supporting examples or the word level description of the regions. The third is a baseline condition (No-teaching+AI-prediction) that makes the following modifications to the experimental design: the participants skip the teaching stage (Stage 1) and immediately proceed to the testing phase (Stage 2). However, during the testing phase, the participants *can see the AI prediction* before they press the use AI button which gives them an edge compared to the teaching condition. We recruited 50 US based participants from Amazon Mechanical Turk per each condition (150 total). Participants in the non-teaching baseline were paid \$3 for 10 minutes of work and those in the teaching condition received \$6 for 20 minutes of work. The simulated AI had $k_p = 11$ and was randomly chosen to have probability of error 0 or 1 on each cluster. To obtain the 9 teaching examples we run GREEDY-SELECT with the consistent radius strategy with no knowldge of g_0 or h. The examples in the testing phase was obtained first by filtering the data using K-medoids with $K = 200$ as a way to get diverse questions. Then each participant received 7 random questions from the filtered set on which the AI was correct and 8 on which the AI is incorrect. This study was approved by the IRB.

Observations. *Teaching enables participants to better know when not to defer, but not when to defer.* The first three columns of Table 2 display the metrics measured across both conditions on all participants. We can first note that participants with teaching are able to predict overall just as well as participants in the baseline no-teaching condition who have additional information about the AI prediction at test time. Moreover, participants who received teaching can better recognize when they are able to predict better than the AI. There is a difference significant at p-value 0.05 ($t = 2.9$, from a two sample t-test) of the F1 score when the human doesn't defer between our method and the no-teaching baseline and significant at p-value 0.001 ($t = 3.2$) compared to LIME. However, the participants in the teaching condition deferred to the AI when it was incorrect more often than those in the no-teaching baseline condition. A positive difference significant at p-value 0.05 ($t = -2.0$) in F1 when the humans defers for No-teaching+AI-prediction workers. An explanations for this is that the participants might press the use AI button on examples where their own prediction agrees with that of the AI instead of manually selecting the answer which takes more effort.

Accurate teaching lessons might predict improved task performance and our method teaches more participants than LIME. Given our knowledge about the clusters and the AI, the correct form of the teaching lesson of each example is "AI is good/bad at TOPIC" where TOPIC designates the theme of each cluster amongst a set of 11 topics. Manually inspecting the lessons of the 50 participants without seeing their test performance, we found that 25 out of 50 participants in our teaching condition were able to properly extract the right lesson from each teaching example. The remaining 25 participants were split into two camps: those who gave explanations on question/answer type or too broad or narrow of explanations e.g. "AI is good at people" rather than a specific subgroup of musicians for example (14 out of 50), and those who gave non-comprehensible explanations (11 out of 50, this group performed non trivially and so could not be disqualified). Results for participants who had accurate vs not accurate lessons are shown in the last four columns of Table 2. The participants who had accurate lessons had a



Passage:

Nothoscordum is a genus of New World plants in the onion tribe within the Amaryllis family. It is probably paraphyletic. [...] .
Callirhoe is a genus of flowering plants in the mallow family, Malvaceae. Its nine species are commonly known as poppy mallows and all are native to the prairies and grasslands of North America. Of the nine, some are annuals while others are perennial plants.

Question:

Which genus is native to more continents, Nothoscordum or Callirhoe?

Select your own answer or use the AI's answer:

AI lessons when correct:
Good at soccer but not other sports
AI lessons when incorrect:
bad at chemistry and flowers, but not on geology or geography.

(a) Testing interface

AI is incorrect
For all examples as similar as this.

Passage:

Citric acid is a weak organic tricarboxylic acid having the chemical formula CHo. In biochemistry, it is an intermediate in the citric acid cycle, which occurs in the metabolism of all aerobic organisms. A clementine ("citrus × clementina") is a hybrid between a mandarin orange and a sweet orange, so named in 1902. The exterior is a deep orange colour with a smooth, glossy appearance. clementines can be separated into 7 to 14 segments. [...]

Question:

What is the formula of the organic material that clementines have less of than oranges?

AI is correct
For examples not as similar.

Passage:

The ogallala aquifer is a shallow water table aquifer surrounded by sand, silt, clay and gravel located beneath the great plains in the united states. one of the world's largest aquifers. It was named in 1898 by geologist N. H. Darton from its type locality near the town of ogallala, Nebraska.
[...] . The population was 4,737 at the 2010 census. It is the county seat of keith county.

Question:

What shallow water table aquifer is located near the county seat of Keith County, Nebraska?

The following words are most representative of this example and its surrounding:
subspecies, fabaceae, genus of, shrubs, plant, plants, species, flowering, genus

Write a sentence to describe the theme of the example you solved, be inspired by the words above and the two supporting examples.

AI is bad at chemistry and flowers, but not on geology or geography.

(b) Teaching interface

Figure 3: On the left in subfigure (a) is the testing interface shown for an example. This is the same interface that is also shown at the beginning of each teaching example. After the human predicts and we are in the teaching phase, we show them the correct answer and transition to the interface in subfigure (b) that shows the two supporting examples for the example in (a), the top weighted words in the region and asks the user to write down their rule for the example.

Metric	Ours-Teaching (all)	No-Teaching	LIME (all)	Ours (acc)	Ours (inacc)	LIME (acc)	LIME (inacc)
Overall F1	58.2 ± 3.4	57.6 ± 3.4	52.9 ± 3.4	62.8 ± 4.7	53.5 ± 4.9	56.5 ± 6.4	52.0 ± 4.2
Defer F1	50.7 ± 4.7	57.8 ± 4.9	48.1 ± 5.3	53.4 ± 6.7	50.0 ± 6.8	44.6 ± 9.0	49.9 ± 6.5
Non-Defer F1	67.6 ± 4.7	57.6 ± 4.7	56.9 ± 4.6	73.92± 6.2	60.6 ± 7.1	70.0 ± 8.6	53.7 ± 5.4
Time/ex (min)	0.60 ± 0.03	0.62 ± 0.03	0.68 ± 0.04	0.54 ± 0.04	0.68 ± 0.05	0.65 ± 0.08	0.69 ± 0.05
AI-Reliance (%)	55.2 ± 3.6	48.9 ± 3.6	45.4 ± 3.6	53.3 ± 4.9	58.9 ± 5.0	52.8 ± 3.6	43.6 ± 4.3

Table 2: Comparison of the metrics between our teaching condition (split into all participants, those who gave accurate lessons (acc) and those who didn't (inacc), see description below), the `No-teaching+AI-prediction` condition and LIME teaching. Shown are averages across all participants with 95% confidence interval error bars. The F1 of the AI alone in this setting is 46.7%; we did not separately measure the F1 of the human in isolation.

9 point average overall F1 difference significant at p-value 0.01 compared to those with inaccurate lessons. With LIME-Teaching we found that only 14 out of 50 participants were able to properly extract the right lessons. The difference between LIME and our method in enabling teaching is significant at p-value 0.02 with $t = 2.3$, however, we observe that accurate teaching has a similar effect in both conditions. Note, that even when participants have accurate lessons, they often don't always follow their own recommendations as evidenced by the low Defer F1 score.

Additional Synthetic Experiment

To complement our NLP-based experiments, we run a study on CIFAR-10 (Krizhevsky, Hinton et al. 2009) consisting of images from 10 classes. We train a WideResNet model with no data augmentation as the AI (Zagoruyko and Komodakis 2016). The message the AI sends is the pair $A = (\hat{y}, \hat{c})$ consisting of the prediction $\hat{y}$ and confidence score $\hat{c}$ (softmax output of the model). We use the human expert model from Mozannar and Sontag (2020): if the image is in the first 6 classes, the expert is perfect, otherwise the expert predicts randomly. During teaching, we assume the human learns according to Assumption 2 and uses the radius given by the teaching set. The human's prior is to ignore the AI if $\hat{c}$ is less than .5. We find that with only 4 teaching examples, `DOUBLE-GREEDY` increases accuracy from 90.98 to 96.3 ± 0.1 on the test set. Additional results are in the appendix.

Discussion

One limitation of our experiments is that we used a simulated AI that has an easier to understand error boundary. This enabled us to have a more in-depth study of the crowdworker responses than otherwise would have been possible. Another limitation is that our test interface did not include model explanations, which was done to eliminate additional confounding factors when comparing approaches. Future work will remedy both limitations. Teaching is used in our work to influence a human's perception of an AI model; this could potentially be misused if the AI predictions are not faithful.

Acknowledgments

HM and DS were supported by NSF AitF award CCF-1723344

References

Amershi, S.; Weld, D.; Vorvoreanu, M.; Fourney, A.; Nushi, B.; Collisson, P.; Suh, J.; Iqbal, S.; Bennett, P. N.; Inkpen, K.; et al. 2019. Guidelines for human-AI interaction. In *Proceedings of the 2019 chi conference on human factors in computing systems*, 1–13.

Atkinson, R. K.; Derry, S. J.; Renkl, A.; and Wortham, D. 2000. Learning from examples: Instructional principles from the worked examples research. *Review of educational research*, 70(2): 181–214.

Bansal, G.; Nushi, B.; Kamar, E.; Horvitz, E.; and Weld, D. S. 2021. Is the Most Accurate AI the Best Teammate? Optimizing AI for Teamwork. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 11405–11414.

Bansal, G.; Nushi, B.; Kamar, E.; Lasecki, W. S.; Weld, D. S.; and Horvitz, E. 2019. Beyond Accuracy: The Role of Mental Models in Human-AI Team Performance. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 7, 2–11.

Bornstein, A. M.; Khaw, M. W.; Shohamy, D.; and Daw, N. D. 2017. Reminders of past choices bias decisions for reward in humans. *Nature Communications*, 8(1): 1–9.

Bourgin, D. D.; Peterson, J. C.; Reichman, D.; Russell, S. J.; and Griffiths, T. L. 2019. Cognitive model priors for predicting human decisions. In *International conference on machine learning*, 5133–5141. PMLR.

Cai, C. J.; Winter, S.; Steiner, D.; Wilcox, L.; and Terry, M. 2019. " Hello AI": Uncovering the Onboarding Needs of Medical Practitioners for Human-AI Collaborative Decision-Making. *Proceedings of the ACM on Human-computer Interaction*, 3(CSCW): 1–24.

Chandrasekaran, A.; Prabhu, V.; Yadav, D.; Chattopadhyay, P.; and Parikh, D. 2018. Do explanations make VQA models more predictable to a human? *arXiv preprint arXiv:1810.12366*.

DeVries, T.; and Taylor, G. W. 2018. Learning confidence for out-of-distribution detection in neural networks. *arXiv preprint arXiv:1802.04865*.

Feng, S.; and Boyd-Graber, J. 2019. What can AI do for me? evaluating machine learning interpretations in cooperative play. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, 229–239.

Gaube, S.; Suresh, H.; Raue, M.; Merritt, A.; Berkowitz, S. J.; Lermer, E.; Coughlin, J. F.; Guttag, J. V.; Colak, E.; and Ghassemi, M. 2021. Do as AI say: susceptibility in deployment of clinical decision-aids. *NPJ digital medicine*, 4(1): 1–8.

Giguère, G.; and Love, B. C. 2013. Limits in decision making arise from limits in memory retrieval. *Proceedings of the National Academy of Sciences*, 110(19): 7613–7618.

Graves, A.; Bellemare, M. G.; Menick, J.; Munos, R.; and Kavukcuoglu, K. 2017. Automated curriculum learning for neural networks. In *international conference on machine learning*, 1311–1320. PMLR.

Hattie, J.; and Timperley, H. 2007. The power of feedback. *Review of educational research*, 77(1): 81–112.

Hunziker, A.; Chen, Y.; Mac Aodha, O.; Rodriguez, M. G.; Krause, A.; Perona, P.; Yue, Y.; and Singla, A. 2018. Teaching multiple concepts to a forgetful learner. *arXiv preprint arXiv:1805.08322*.

Ilvento, C. 2019. Metric learning for individual fairness. *arXiv preprint arXiv:1906.00250*.

Kaur, H.; Nori, H.; Jenkins, S.; Caruana, R.; Wallach, H.; and Wortman Vaughan, J. 2020. Interpreting Interpretability: Understanding Data Scientists' Use of Interpretability Tools for Machine Learning. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 1–14.

Kim, Y.-S.; Walls, L. A.; Krafft, P.; and Hullman, J. 2019. A bayesian cognition approach to improve data visualization. In *Proceedings of the 2019 CHI conference on human factors in computing systems*, 1–14.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images. *Citeseer*.

Lai, V.; Liu, H.; and Tan, C. 2020. " Why is' Chicago'deceptive?" Towards Building Model-Driven Tutorials for Humans. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 1–13.

Lai, V.; and Tan, C. 2019. On human predictions with explanations and predictions of machine learning models: A case study on deception detection. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 29–38.

Link, D.; Hellingrath, B.; and Ling, J. 2016. A Human-is-the-Loop Approach for Semi-Automated Content Moderation. In *ISCRAM*.

Madras, D.; Pitassi, T.; and Zemel, R. 2018. Predict Responsibly: Improving Fairness and Accuracy by Learning to Defer. In *Advances in Neural Information Processing Systems*, 6150–6160.

Mozannar, H.; Satyanarayan, A.; and Sontag, D. 2021. Teaching Humans When To Defer to a Classifier via Examplars. arXiv:2111.11297.

Mozannar, H.; and Sontag, D. 2020. Consistent estimators for learning to defer to an expert. In *International Conference on Machine Learning*, 7076–7087. PMLR.

Qi, G.-J.; Tang, J.; Zha, Z.-J.; Chua, T.-S.; and Zhang, H.-J. 2009. An efficient sparse metric learning in high-dimensional space via l 1-penalized log-determinant regularization. In *Proceedings of the 26th Annual International Conference on Machine Learning*, 841–848.

Raghu, M.; Blumer, K.; Corrado, G.; Kleinberg, J.; Obermeyer, Z.; and Mullainathan, S. 2019. The algorithmic automation problem: Prediction, triage, and human effort. *arXiv preprint arXiv:1903.12220*.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Reimers, N.; and Gurevych, I. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. " Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 1135–1144.

Richler, J. J.; and Palmeri, T. J. 2014. Visual category learning. *Wiley Interdisciplinary Reviews: Cognitive Science*, 5(1): 75–94.

Shaikh, S. J.; and Cruz, I. 2019. 'Alexa, Do You Know Anything?'The Impact of an Intelligent Assistant on Team Interactions and Creative Performance Under Time Scarcity. *arXiv preprint arXiv:1912.12914*.

Singla, A.; Bogunovic, I.; Bartok, G.; Karbasi, A.; and Krause, A. 2014. Near-Optimally Teaching the Crowd to Classify. In *International Conference on Machine Learning*, 154–162.

Suresh, H.; Lewis, K. M.; Guttag, J. V.; and Satyanarayan, A. 2021. Intuitively Assessing ML Model Reliability through Example-Based Explanations and Editing Model Inputs. *arXiv preprint arXiv:2102.08540*.

Tu, M.; Huang, K.; Wang, G.; Huang, J.; He, X.; and Zhou, B. 2020. Select, answer and explain: Interpretable multi-hop reading comprehension over multiple documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 9073–9080.

Wilder, B.; Horvitz, E.; and Kamar, E. 2020. Learning to Complement Humans. *arXiv preprint arXiv:2005.00582*.

Wortman Vaughan, J.; and Wallach, H. 2021. A Human-Centered Agenda for Intelligible Machine Learning. This is a draft version of a chapter in a book to be published in the 2020 - 21 timeframe.

Yang, Z.; Qi, P.; Zhang, S.; Bengio, Y.; Cohen, W.; Salakhutdinov, R.; and Manning, C. D. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2369–2380.

Zagoruyko, S.; and Komodakis, N. 2016. Wide residual networks. *arXiv preprint arXiv:1605.07146*.