Teaching Humans When To Defer to a Classifier via Exemplars

HUSSEIN MOZANNAR, Massachusetts Institute of Technology, USA
ARVIND SATYANARAYAN, Massachusetts Institute of Technology, USA
DAVID SONTAG, Massachusetts Institute of Technology, USA

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 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.

CCS Concepts: • **Human-centered computing** → **HCI theory, concepts and models**; **Empirical studies in HCI**; • **Computing methodologies** → *Machine learning*.

Additional Key Words and Phrases: human mental models, onboarding, teaching, example selection, question answering

ACM Reference Format:
Hussein Mozannar, Arvind Satyanarayan, and David Sontag. 2022. Teaching Humans When To Defer to a Classifier via Exemplars. 1, 1 (April 2022), 12 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Automated agents powered by machine learning are augmenting the capabilities of human decision makers in settings such as healthcare [5, 11], content moderation [23] and more routine decisions such as asking AI-enabled virtual assistants for recommendations [32]. 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. We propose that before an AI agent is deployed to assist a human decision maker, the human is taught through a tailored onboarding phase how to make decisions with the help of the AI. The purpose of the onboarding is to help the human understand when to trust the AI and how the AI can complement their abilities. 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.

Authors' addresses: Hussein Mozannar, mozannar@mit.edu, Massachusetts Institute of Technology, USA; Arvind Satyanarayan, Massachusetts Institute of Technology, USA; David Sontag, Massachusetts Institute of Technology, USA.

Our onboarding phase consists of letting the human predict on a series of specially selected teaching examples in a setting that mimics the deployment use case. The examples are chosen to give an overview of the AI's strengths and weaknesses especially when it complement's the abilities of the human. After predicting on each example, the human agent then receives feedback on their performance and that of the AI. To allow the human to generalize from each example, we display features of the region surrounding the example. Finally, to enable retention of the example, we let the human write down a lesson indicating whether they should trust the AI in that region and what characterizes the region. Our approach is inspired by research in the education literature that highlight the importance of feedback and lesson retention for learning [2, 15].

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* in section 3. 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 in section 4. This novel parameterization is inspired by work in cognitive science that suggests that humans make decisions by weighing similar past experiences [6]. Assuming this rejector model, we give a greedy strategy for selecting a set of representative teaching examples that allows us to control the examples and the region surrounding them in section 5.

For our main evaluation, we conduct experiments on Amazon Mechanical Turk on the task of passage-based question answering from HotpotQA [39] in section 6. 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.

2 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 [13, 14, 19, 20, 22, 33–35, 38, 41]. 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 [17]. 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. Another direct approach for teaching is presenting the human with a set of guidelines of when to rely on the AI [1]. However, these guidelines need to be developed by a set of domain experts and no standard approach currently exists for creating such guidelines.

The reverse setting, of teaching a classifier when to defer to a human, is dubbed as learning to defer (LTD) [24, 25, 27, 37]. 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.

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 [21, 30]. Their selection strategy does not take into account the human predictor, nor does their approach do more than display the examples. On a task of visual question answering, [9] handpicked 7 examples to teach crowdworkers about the AI abilities and found that teaching improved the ability to detect the AI's failure. [10] on a Quizbowl question answering task highlight the importance of modeling the skill level of the human expert when

designing the explanations; this further motivates our incorporation of the human predictor into the choice of the teaching set. Through a study of 21 pathologists, [8] gathered a set of guidelines of what clinicians wanted to know about an AI prior to interacting with it. [40] study the effect of initial debriefing of stated AI accuracy compared to observed AI accuracy in deployment and find a significant effect of stated accuracy on trust, but that diminishes quickly after observing the model in practice; this reinforces our approach of building trust through examples that simulate deployment. [4] 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. [3] 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.

3 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:

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

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:

$$L(\pi, \pi_Y, h, r) = \mathbb{E}_{x,z,y}[\underbrace{l(\pi_Y(x), y)}_{\text{AI cost}} \overbrace{\mathbb{I}_{r(x,\pi(x))=1}}^{\text{AI predicts}} + \underbrace{l(h(z, \pi(x)), y)}_{\text{Human cost}} \overbrace{\mathbb{I}_{r(x,\pi(x))=0}}^{\text{Human predicts}}] \tag{2}$$

We put ourselves in the role of a system designer who has knowledge of both the human and the AI and wishes to minimize the loss of the system L (2).

The central Human-AI interaction problem. Given a fixed AI policy, and fixed 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 AI 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.

4 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*, where we define domain experts to mean 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 [6, 12, 31]. Moreover, [6] 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 $g_0(Z, A) : \mathcal{Z} \times \mathcal{A} \rightarrow \{0, 1\}$ 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 $K(.,.) : \mathcal{Z} \times \mathcal{Z} \rightarrow \mathbb{R}^+$ 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:

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

where $B(Z)$ is the set of all points in D_T that they observed in training sufficiently similar to Z:

$$B(Z) = \{i \in [m] \mid K(Z, z_i) > \gamma_i\} \tag{4}$$

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:

$$vote(B(Z)) = \arg \max_{k \in \{0,1\}} \frac{\sum_{i \in B(Z)} \mathbb{I}\{r_i = k\} K(Z, z_i)}{\sum_{i \in B(Z)} K(Z, z_i)} \tag{5}$$

Where r_i is the deferral rule that the human has learned on example z_i.

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 $K(,,.)$. The prior rejector g_0 can be learned by testing the human prior as evidenced by prior work on capturing human priors [7, 18], otherwise, a

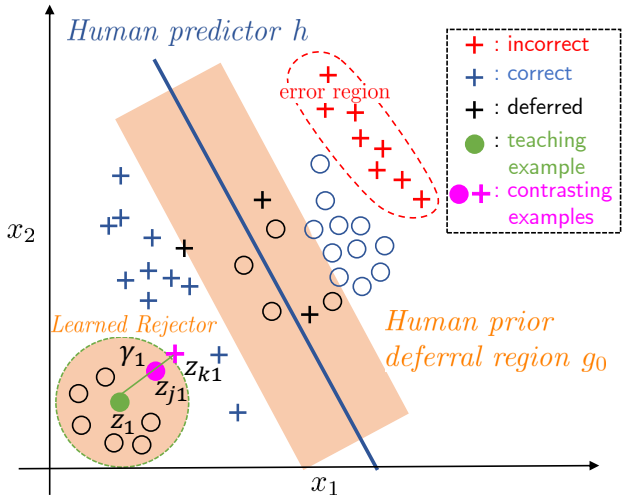


Fig. 1. Illustration of human rejector on toy example. The task is classification with labels $\{o, +\}$, the human prediction h is the blue line and the prior g_0 is the shaded orange region surrounding the boundary. Points in red is where the human is incorrect, in blue correct and in black point deferred 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.

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 [16, 26], or rely on an AI based similarity measure e.g. from neural network embeddings [29]. 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.

5 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. Essentially, our approach is trying to find local regions, balls with respect to $K(.,.)$, that best teach the human about the AI. 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:

$$r_i = \mathbb{I}\{\mathbb{E}[l(h(z_i, a_i), y_i)] \geq \mathbb{E}[l(\pi_Y(x_i), y_i]]\} \tag{6}$$

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

Algorithm 1 Our Human Teaching Approach

1: **Input**: Teaching set D
2: **for** $i = 1, \cdots, m$ **do**
3: **Stage 1: Testing.** Test the human on example z_i with AI message a_i
4: **Stage 2: Feedback.** Show human feedback of actual label y_i, AI prediction π_i, and recommended deferral action r_i.
5: **Stage 3: Lesson Generalization.** Show the two contrasting examples z_j and z_k and high level features about the region to allow generalization around z_i. Enables learning of the radius γ_i.
6: **Stage 4: Lesson Reinforcement.** We ask the human to write a rule R_i that describes the region surrounding the example z_i and which action they should take. They can rely on this rule in the future and allows for reflection.
7: **end for**

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:

$$L(D) = \sum_{i \in S} l\left(M(z_i, a_i; D), y_i\right) \tag{7}$$

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:

$$z, \gamma = \arg \min_{z_i \in S \setminus D_t, \gamma} L(D_t \cup \{z_i, \gamma\}), \tag{8}$$

$$\text{s.t. } \exists k \in [n] \; s.t. \; \gamma = K(z_i, z_k), \tag{9}$$

$$\text{and } \frac{\sum_{j \in [n], K(z_i, z_j) > \gamma} \mathbb{I}_{r_j = r_i}}{|\{j \in [n], K(z_i, z_j) > \gamma\}|} \geq \alpha \tag{10}$$

Constraint (9) restricts γ to be the similarity between z and another data point and constraint (10) 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`.

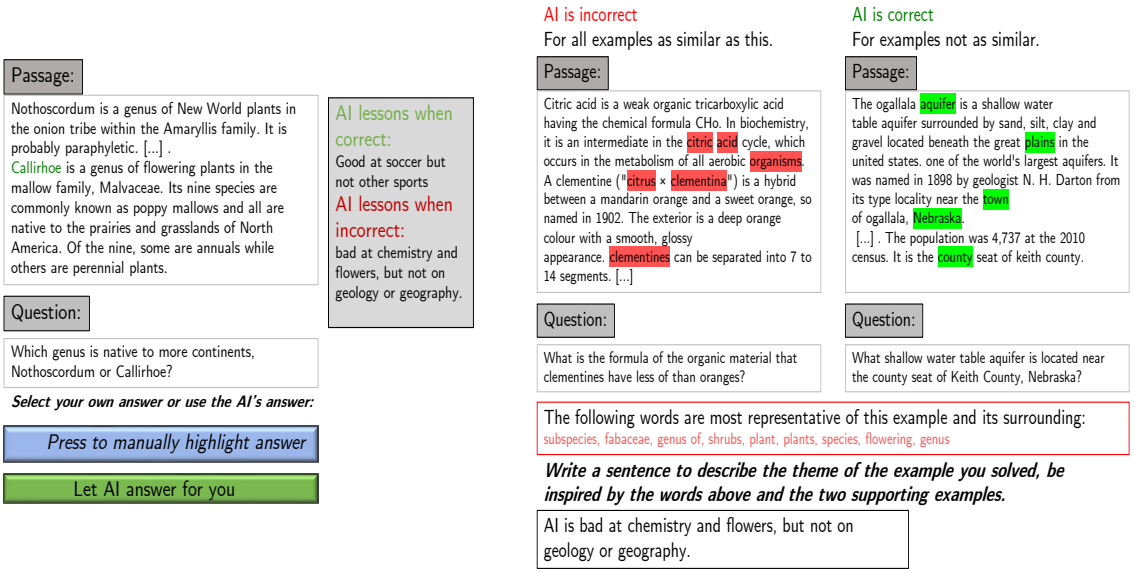
Contrasting examples. Note that the radius γ is actually defined by two points: the point z_k in equation (9) that defines the boundary and an interior point z_j that is the least similar point to z with similarity at least γ; these two points are illustrated in Figure 1 with the color pink. These two 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.

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 shown in Algorithm 1.

6 EXPERIMENTAL USER STUDY

6.1 Experimental Preliminaries

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. We rely on the



(a) Testing interface (b) Teaching interface

Fig. 2. 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.

HotpotQA dataset [39] 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. 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.

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 [28]; 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 2a.

Testing user interface. Our user interface during testing is shown in Figure 2a 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.

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 1. 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.

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 [30] (see Figure 2b). 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. The experimental teaching setup proceeds in three stages. The first stage (Stage 0) is a tutorial that introduces the task with two examples and where we gather the worker's demographic information, knowledge of machine learning and how often they visit Wikipedia. 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. After the two stages is an exit survey where users are asked about their decision process for using the AI. We randomly assign each participant to one of three conditions.

Experimental 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.

Participants. 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. Any demographic information we gathered in our study is kept confidential and workers were asked to consent to their use of their responses in research studies.

Simulated AI. One of the top performing models on HotpotQA is SAE-large: a graph neural network on top of RoBERTa embeddings [36]. We performed a detailed error analysis 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 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 11 clusters based on only the paragraph embeddings obtained from a pre-trained SentenceBERT model [29] and was randomly chosen to have probability of error 0 or 1 on each cluster. The answer of the AI when it is incorrect is manually constructed to be reasonably incorrect: for example if the answer asks for a date, we provide an incorrect date rather than a random sentence. To summarize, the AI for each cluster in the data has a specified probability of error that is constant on the cluster. To show that each cluster computed has a distinct meaningful theme, we retrieve the top 10 most common Wikipedia categories in each cluster. Example cluster categories include singers/musicians, movies and soccer (but not football).

Teaching Set. To obtain the 9 teaching examples we run `GREEDY-SELECT` with the consistent radius strategy with no knowledge of g_0 or h (both assumed to be equally likely to predict 1 or 0 on each point). 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 is calibrated to the AI model actual error rate.

6.2 User Study Observations and Results

Teaching enables participants to better know when to predict on their own, but not when to defer to the AI. The first three columns of Table 1 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. A

User Lessons. In Table 2 we show examples of the lessons that the crowdworkers wrote during the teaching phase for the proposed teaching method. We show examples of the lessons on the first 2 examples in the teaching phase and separate the participant lessons into 4 categories: participants who wrote accurate lessons, participants who wrote irrelevant lessons (not relevant to the question or required no effort to write), participants who wrote complex lessons that don't pertain to the example topic and finally participants who wrote narrow lessons that are on topic but only apply to the example and not the neighborhood of the example.

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 which include soccer, politics, music and more. 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 irrelevant 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 1. The participants who had accurate lessons had a 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.

Table 2. Example of lessons that users in the Ours-Teaching condition wrote during the teaching phase. We show examples of the lessons on the first 2 examples in the teaching phase and separate the participant lessons into 4 categories.

Lesson Type	Example ID	Actual Lesson
Accurate Lessons	1	The AI is not good at answering questions about plants.
Accurate Lessons	2	The AI is better at Politics and geography than at sports.
Irrelevant Lessons	1	I understood AI is good at answering
Irrelevant Lessons	2	AI focus on the institution
Complex Lessons	1	It seems to be better at answering questions where the absolute same phrases are used in the question as the passage and where both answers are in the question, maybe?
Complex Lessons	2	The ai is good at answering questions that has to do with cities and numbers though not good with words that has to do with repeated words.
Narrow Lessons	1	The AI isn't good at multi-faceted questions about continental species.
Narrow Lessons	2	The topic was politics and the AI is good at answering questions about specific areas when the question can be answered by looking for specific information about one section but not when it involves integrating multiple pieces of information from the paragraph.

7 DISCUSSION

One limitation of our human 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. Having a simulated AI which we perfectly understand where its error regions are, enables us to define what the "lessons" should be and thus evaluate if users are learning correctly. Future user studies will evaluate with non-simulated AI models. We hypothesize that the example selection algorithm presented in this work will be sufficient, however, we might require better methods to illustrate the neighborhood for each example. Another limitation is that our test-time interface did not include model explanations, which was done to eliminate additional confounding factors when comparing approaches. Future work will evaluate whether the effect of teaching remains as significant when evaluating with test-time model explanations.

Teaching is used in our work to influence a human's perception of an AI model; this could potentially be misused to manipulate humans into relying on AI agents if the AI predictions during teaching were fabricated. In the absence of this, we believe our approach can serve to give an unbiased overview of the AI.

REFERENCES

[1] Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul N Bennett, Kori Inkpen, et al. 2019. Guidelines for human-AI interaction. In *Proceedings of the 2019 chi conference on human factors in computing systems*. 1–13.

[2] Robert K Atkinson, Sharon J Derry, Alexander Renkl, and Donald Wortham. 2000. Learning from examples: Instructional principles from the worked examples research. *Review of educational research* 70, 2 (2000), 181–214.

[3] Gagan Bansal, Besmira Nushi, Ece Kamar, Eric Horvitz, and Daniel S Weld. 2021. Is the Most Accurate AI the Best Teammate? Optimizing AI for Teamwork. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 11405–11414.

[4] Gagan Bansal, Besmira Nushi, Ece Kamar, Walter S Lasecki, Daniel S Weld, and Eric Horvitz. 2019. Beyond Accuracy: The Role of Mental Models in Human-AI Team Performance. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, Vol. 7. 2–11.

[5] Emma Beede, Elizabeth Baylor, Fred Hersch, Anna Iurchenko, Lauren Wilcox, Paisan Ruamviboonsuk, and Laura M Vardoulakis. 2020. A Human-Centered Evaluation of a Deep Learning System Deployed in Clinics for the Detection of Diabetic Retinopathy. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–12.

[6] Aaron M Bornstein, Mel W Khaw, Daphna Shohamy, and Nathaniel D Daw. 2017. Reminders of past choices bias decisions for reward in humans. *Nature Communications* 8, 1 (2017), 1–9.

[7] David D Bourgin, Joshua C Peterson, Daniel Reichman, Stuart J Russell, and Thomas L Griffiths. 2019. Cognitive model priors for predicting human decisions. In *International conference on machine learning*. PMLR, 5133–5141.

[8] Carrie J Cai, Samantha Winter, David Steiner, Lauren Wilcox, and Michael Terry. 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 (2019), 1–24.

[9] Arjun Chandrasekaran, Viraj Prabhu, Deshraj Yadav, Prithvijit Chattopadhyay, and Devi Parikh. 2018. Do explanations make VQA models more predictable to a human? *arXiv preprint arXiv:1810.12366* (2018).

[10] Shi Feng and Jordan Boyd-Graber. 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.

[11] Susanne Gaube, Harini Suresh, Martina Raue, Alexander Merritt, Seth J Berkowitz, Eva Lermer, Joseph F Coughlin, John V Guttag, Errol Colak, and Marzyeh Ghassemi. 2021. Do as AI say: susceptibility in deployment of clinical decision-aids. *NPJ digital medicine* 4, 1 (2021), 1–8.

[12] Gyslain Giguère and Bradley C Love. 2013. Limits in decision making arise from limits in memory retrieval. *Proceedings of the National Academy of Sciences* 110, 19 (2013), 7613–7618.

[13] Ana Valeria Gonzalez, Gagan Bansal, Angela Fan, Robin Jia, Yashar Mehdad, and Srinivasan Iyer. 2020. Human Evaluation of Spoken vs. Visual Explanations for Open-Domain QA. *arXiv preprint arXiv:2012.15075* (2020).

[14] Peter Hase and Mohit Bansal. 2020. Evaluating Explainable AI: Which Algorithmic Explanations Help Users Predict Model Behavior? *arXiv preprint arXiv:2005.01831* (2020).

[15] John Hattie and Helen Timperley. 2007. The power of feedback. *Review of educational research* 77, 1 (2007), 81–112.

[16] Christina Ilvento. 2019. Metric learning for individual fairness. *arXiv preprint arXiv:1906.00250* (2019).

[17] Harmanpreet Kaur, Harsha Nori, Samuel Jenkins, Rich Caruana, Hanna Wallach, and Jennifer Wortman Vaughan. 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.

[18] Yea-Seul Kim, Logan A Walls, Peter Krafft, and Jessica Hullman. 2019. A bayesian cognition approach to improve data visualization. In *Proceedings of the 2019 CHI conference on human factors in computing systems*. 1–14.

[19] Rafal Kocielnik, Saleema Amershi, and Paul N Bennett. 2019. Will you accept an imperfect ai? exploring designs for adjusting end-user expectations of ai systems. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–14.

[20] Isaac Lage, Emily Chen, Jeffrey He, Menaka Narayanan, Been Kim, Sam Gershman, and Finale Doshi-Velez. 2019. An evaluation of the human-interpretability of explanation. *arXiv preprint arXiv:1902.00006* (2019).

[21] Vivian Lai, Han Liu, and Chenhao Tan. 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.

[22] Vivian Lai and Chenhao Tan. 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.

[23] Daniel Link, Bernd Hellingrath, and Jie Ling. 2016. A Human-is-the-Loop Approach for Semi-Automated Content Moderation.. In *ISCRAM*.

[24] David Madras, Toni Pitassi, and Richard Zemel. 2018. Predict Responsibly: Improving Fairness and Accuracy by Learning to Defer. In *Advances in Neural Information Processing Systems*. 6150–6160.

[25] Hussein Mozannar and David Sontag. 2020. Consistent estimators for learning to defer to an expert. In *International Conference on Machine Learning*. PMLR, 7076–7087.

[26] Guo-Jun Qi, Jinhui Tang, Zheng-Jun Zha, Tat-Seng Chua, and Hong-Jiang Zhang. 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.

[27] Maithra Raghu, Katy Blumer, Greg Corrado, Jon Kleinberg, Ziad Obermeyer, and Sendhil Mullainathan. 2019. The algorithmic automation problem: Prediction, triage, and human effort. *arXiv preprint arXiv:1903.12220* (2019).

[28] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250* (2016).

[29] Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084* (2019).

[30] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 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.

[31] Jennifer J Richler and Thomas J Palmeri. 2014. Visual category learning. *Wiley Interdisciplinary Reviews: Cognitive Science* 5, 1 (2014), 75–94.

[32] Sonia Jawaid Shaikh and Ignacio Cruz. 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* (2019).

[33] Alison Smith-Renner, Ron Fan, Melissa Birchfield, Tongshuang Wu, Jordan Boyd-Graber, Daniel S Weld, and Leah Findlater. 2020. No Explainability without Accountability: An Empirical Study of Explanations and Feedback in Interactive ML. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–13.

[34] Harini Suresh, Natalie Lao, and Ilaria Liccardi. 2020. Misplaced Trust: Measuring the Interference of Machine Learning in Human Decision-Making. *arXiv preprint arXiv:2005.10960* (2020).

[35] Harini Suresh, Kathleen M Lewis, John V Guttag, and Arvind Satyanarayan. 2021. Intuitively Assessing ML Model Reliability through Example-Based Explanations and Editing Model Inputs. *arXiv preprint arXiv:2102.08540* (2021).

[36] Ming Tu, Kevin Huang, Guangtao Wang, Jing Huang, Xiaodong He, and Bowen Zhou. 2020. Select, answer and explain: Interpretable multi-hop reading comprehension over multiple documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 9073–9080.

[37] Bryan Wilder, Eric Horvitz, and Ece Kamar. 2020. Learning to Complement Humans. *arXiv preprint arXiv:2005.00582* (2020).

[38] Jennifer Wortman Vaughan and Hanna Wallach. 2021. A Human-Centered Agenda for Intelligible Machine Learning. (May 2021). https://www.microsoft.com/en-us/research/publication/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..

[39] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 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.

[40] Ming Yin, Jennifer Wortman Vaughan, and Hanna Wallach. 2019. Understanding the effect of accuracy on trust in machine learning models. In *Proceedings of the 2019 chi conference on human factors in computing systems*. 1–12.

[41] Yunfeng Zhang, Q Vera Liao, and Rachel KE Bellamy. 2020. Effect of confidence and explanation on accuracy and trust calibration in AI-assisted decision making. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 295–305.