Power to the People: The Role of Humans in Interactive Machine Learning

Saleema Amershi, Maya Cakmak, W. Bradley Knox, Todd Kulesza[1]

Abstract

Intelligent systems that learn interactively from their end-users are quickly becoming widespread. Until recently, this progress has been mostly fueled by advances in machine learning; more and more researchers, however, are realizing the importance of studying *users* of these systems. In this article we promote this approach and demonstrate how it can result in better user experiences and more effective learning systems. We present a number of case studies that characterize the impact of interactivity, demonstrate ways in which existing systems fail to account for the user, and explore new ways for learning systems to interact with their users. We argue that the design process for interactive machine learning systems should involve users at all stages: explorations that reveal human interaction patterns and inspire novel interaction methods, as well as refinement stages to tune details of the interface and choose among alternatives. After giving a glimpse of the progress that has been made so far, we discuss the challenges that we face in moving the field forward.

Introduction

Designing machine learning systems is a complex process, requiring input and output identification, feature specification, model and algorithm selection, and parameter tuning. Due to these complexities, the ultimate consumers of machine learning systems (i.e., the *end-users*) have traditionally been shielded from this design process altogether. While this can hide the intricacies of the underlying process, it also limits the end-user's ability to influence the learning system and can lead to undesired behaviors with little to no means for recourse. For example, an end-user may use a machine learning system in a situation or with data never anticipated by the original developer, potentially resulting in unexpected behaviors. In many of these cases, the only way to correct the behavior is to provide feedback to the original developer for the next round of development, which is inefficient and expensive. Moreover, relying on experts to drive such systems prevents end-users from creating their own machine learners to suit their needs and solve their problems.

Take for example the events from the following case study. In 1998, Caruana and his collaborators began work with biochemists to cluster proteins based on their helical structure with the goal of revealing structural insights that could help define a protein taxonomy. While this endeavor helped to shed light on the structural characteristics of proteins, it also took substantially longer than originally anticipated. In his invited talk at the IUI 2013 Workshop on Interactive Machine Learning (Amershi et al. 2013), Caruana recounted this experience as involving a time-consuming cycle. First, machine learning experts would create a clustering and

[1] All authors contributed equally.

accompanying visualizations to summarize that clustering. They would then meet with biochemists (i.e., the domain experts) to discuss the results. At this meeting, the domain experts would critique the clustering, creating constraints (e.g., “these two proteins should / should not be in the same cluster”, “this cluster is too small”). Following each meeting, the machine learning experts would carefully adjust the clustering distance metric to adhere to the given constraints and then re-compute the clusters for the next iteration. In this case study, the machine-learning experts were the only interface available for the domain experts to provide their expertise, resulting in lengthy interaction cycles. Incited by this experience, Caruana *et al*. went on to develop novel feedback techniques for more *interactively* incorporating domain expert knowledge into the distance metric used for clustering (Cohn et al. 2003, Caruana et al. 2006).

Motivated by similar needs and experiences, researchers have recently begun to employ *interactive* machine learning to better leverage end-user knowledge and capabilities during the machine learning process. In the interactive machine learning process, end-users can more directly assess and guide the underlying machine learner in a tighter interactive loop (Figure 1). For example, many commercial recommender systems now employ interactive machine learning to adapt recommendations based on user specified preferences for items (e.g., ‘liking’ or ‘disliking’ items). In each iteration, end-users can inspect new recommendations and then further guide the system by specifying additional preferences.

However, while interactive machine learning is beginning to drive many user-facing applications, until recently much of the progress in this space has been fueled by advances in machine learning. This article advocates that it is equally important to study the *users* of interactive machine learning systems in order to create better user experiences and more effective machine learning. Through a series of case studies, we argue that explicit study of the interaction between humans and machine learners is critical to designing interfaces and machine learning algorithms that facilitate effective interactive machine learning. These case studies also paint a broad picture of the range of recent research on interactive machine learning, serving both as an introduction to the topic and a vehicle for considering the body of research altogether.

We begin by providing a formal definition of interactive machine learning and then illustrate the learning process with archetypal examples that follow a common form of interactive machine learning, in which a user observes learned predictions and then provides further labeled examples informed by those observations. Next, we present research that examines (and often upends) assumptions about end-user interaction with machine learning systems. Concluding the case studies, we review research involving novel interfaces that move beyond the interactions afforded by the archetypal examples, finding that these new techniques often enable more powerful end-user interaction but must be carefully designed so as not to confuse the user or otherwise harm the learning. Finally, we discuss the current state of the field and identify opportunities and open challenges for future interactive machine learning research.

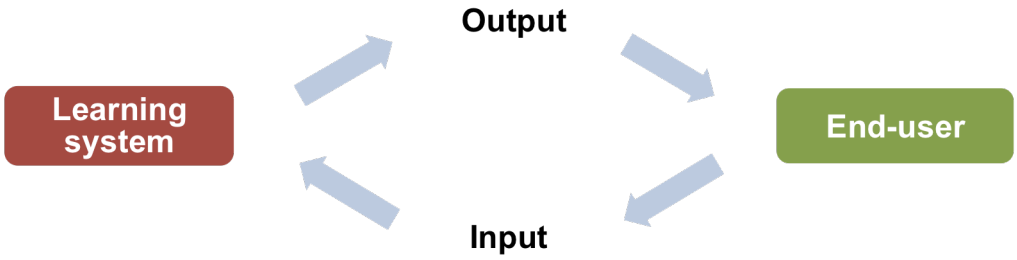

Figure 1: In the interactive machine learning process, a learning system iteratively presents output to a user who can provide new inputs to correct or refine the learning. The learner incorporates this input into its model, allowing the user another opportunity for correction and refinement.

Interactive Machine Learning

We define *interactive machine learning* (IML) as a process that involves a tight interaction loop between a human and a machine learner, where the learner iteratively takes input from the human, promptly incorporates that input, and then provides the human with output impacted by the results of the iteration. This cyclical process is illustrated in Figure 1. In interactive machine learning systems, *learning* is interleaved with *execution*; *i.e.*, the human uses or tests the system while he or she continues to train it. As a result, the output of the system influences the user's subsequent input. An example is a recommendation system such as Pandora[2], which takes labels on played songs as input, and provides new songs that are expected to fit the user's preferences as output. Narrowing this definition further, for a system to be considered an example of interactive machine learning system, we require that the human is *consciously* interacting with the learner in order to improve it. For instance, if a website adapts its webpage presentation to a user's click history *without the user intending to improve the website through these clicks*, this adaptation is not considered as interactive machine learning.

Users of interactive machine learning systems vary. A particularly motivating class of users are domain experts who lack expertise in computer programming or machine learning. However, machine learning experts can also be end-users themselves. For instance, an interface that richly visualizes model error immediately after a change of features or learning parameters would increase interactivity for machine learning experts developing learning systems.

The methods for interfacing with learning systems can also vary widely. Traditionally, the input to such systems has been in the form of labeled examples (e.g., a song labeled as *liked* or *disliked* in Pandora), while the output has been in the form of predicted labels or ratings on new samples (e.g., new songs presented to the user which have high predicted ratings). Recent

[2] www.pandora.com

research, however, has started to explore new interfaces, including interfaces for letting users label parts of items (e.g., Fails & Olsen 2003), adjust model parameters and cost functions (e.g., Kapoor et al. 2010), and directly modify a classifier's features (e.g., Kulesza et al. 2011). We describe some of these different forms of input and output in the "Novel Interfaces for Interactive Machine Learning" section later in this article.

A key property of interactivity is that the tight interaction loop allows the output of the system to influence subsequent user inputs to the system. For example, after observing that labeling a song as *liked* results in recommendations from the same artist, the user may label more songs from other artists to diversify their recommendations. We next present two case studies that exemplify our definition of interactive machine learning.

Interactive image segmentation

Some of the earliest work in this area came from Fails and Olsen (2003), who introduced the term *interactive machine learning*. Similar to our definition, they highlighted the *train-feedback-correct* cycle—a process in which the user iteratively provides *corrective* examples to the learner after viewing its output. Their system, called Crayons, allowed users to train a pixel classifier by iteratively marking pixels as foreground or background through brushstrokes on an image. After each user interaction, the system responded with an updated image segmentation for further review and corrective input by the user. Evaluations of the Crayons system via user studies revealed that this immediate output allowed users to instantly perceive misclassifications and correct them by adding new training data in the most problematic areas. As illustrated in Figure 2, after an initial classification, the user provides Crayons with more data at the edges of the hand where the classifier failed. When asked what they were thinking while interacting with the system, most users stated that they were focused on seeing parts of the image that were classified incorrectly.

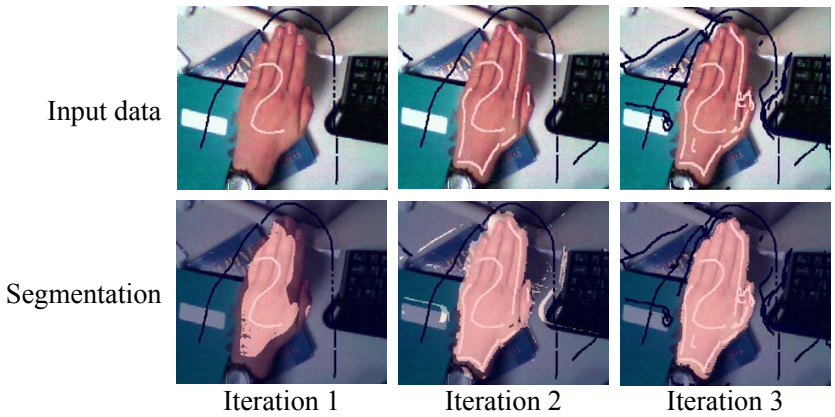

Figure 2: Interactive training of the Crayons system (Fails & Olsen 2003). The system takes pixels labeled as background/foreground as input (provided through brush strokes), and gives a fully segmented image as output (obtained through a classifier that labels each pixel as foreground/background). The user's input is focused on areas where the classifier is failing in previous iterations.

Fails and Olsen's work on Crayons demonstrated that users modify their behavior based on a learner's outputs, which is an underlying premise for much of the following research on interactive machine learning.

Musicians training instruments and instruments training musicians

The realm of music composition and performance is naturally interactive—musicians are accustomed to receiving immediate auditory feedback when interacting with a musical instrument. Fiebrink and colleagues (2011) developed the Wekinator, a machine learning system for enabling people to interactively create novel gesture-based instruments. For example, moving an arm in front of a web cam could produce different sounds based on the arm's position, speed, or rotation. In this system, a neural network receives paired gestures and sounds from the user as input and then learns how to interpolate from unobserved gesture positions to a range of sounds. Users evaluate their instruments directly by gesturing and assessing the produced sounds.

While exploring the use of Wekinator by students in an interdisciplinary music and computer science course, the authors found that as the participants were training their respective learners, the learners were also training the participants. For example, participants learned how to recognize noise in their training samples and provide clearer examples to the learner. In some cases, participants even adjusted their goals to match the observed capabilities of the learner. In a follow-up investigation with a professional cellist (Fiebrink et al. 2011), the cellist identified long-standing flaws in her playing technique while trying to train a learner; the gesture recognizer revealed her bowing articulation was not as precise as she had believed it to be. By observing the outputs of the system in real-time, users were able to modify their behavior in ways that allowed them to create instruments to their satisfaction.

Summary

These two examples illustrate the interactive machine learning process, in which users observe the outputs of the learning system and then provide further input influenced by those observations. This observe-then-train cycle is fundamental to interactive machine learning. However, many of the case studies to follow will consider less traditional types of input and output, moving beyond labeled examples and observations of learner predictions. The case studies presented in this section also demonstrate the benefits of interactivity, which we will continue to highlight throughout this article.

Studying User Interaction with Interactive Machine Learning

The previous section described the general interactive machine learning process; in this section, we turn to case studies illustrating the importance of understanding how end-users can and do interact with interactive machine learning systems and how such understanding can ultimately lead to better-informed designs. First, we present two case studies that demonstrate how people may violate assumptions made by traditional machine learners about their *input*, resulting in unexpected outcomes and user frustration. The next two case studies indicate that people may *want* to interact with machine learning systems in richer ways than developers anticipated, suggesting changes to the *input* constraints that are built into the *interface*. Finally,

we present a case study that shows that people may desire more transparency about how machine learning systems work—changing the *output* constraints of the *interface*—and that such transparency can improve both the user experience and the resulting models.

People dislike being the oracle for active learning

Active learning is a machine-learning paradigm in which the learner chooses the examples from which it will learn (Settles 2010). These examples are selected from a pool of unlabeled samples based on some selection criterion (e.g., examples for which the learner has maximum uncertainty). The learner then queries an oracle, requesting a label for each selected example. This method has had tremendous success in accelerating learning (i.e., requiring fewer labels to reach a target accuracy) in applications like text classification and object recognition, where multiple oracles are paid to provide labels over a long period of time. However, as Cakmak and colleagues (2010) discovered, when applied to interactive settings such as a person teaching a task to a robot by example, active learning can cause several problems.

Figure 3: Users teaching new concepts to a robot by providing positive and negative examples. (Left) Passive learning: examples are chosen and presented by the user. (Right) Active learning: examples are requested by the learner. Although active learning results in faster convergence, users get frustrated from having to answer the learner's long stream of questions and not having control over the interaction.

Cakmak's study (Figure 3) found that the constant stream of questions from the robot learner during interaction was perceived by the user as imbalanced and annoying. The stream of questions also led to a decline in the user's mental model of how the robot learned, causing some participants to "turn their brain off" or "lose track of what they were teaching" (according to their self report) (Cakmak et al. 2010). Similar findings were reported by Guillory and Bilmes (2011) for Netflix's "active" recommendation system for movies. These studies reveal that humans are not necessarily willing to be simple oracles (i.e., repeatedly telling the computer whether it is right or wrong), breaking a fundamental assumption of active learning. Instead, these systems need to account for human factors such as interruptibility or frustration when employing methods like active learning.

People are biased towards giving positive feedback to learning agents

In reinforcement learning, a robot or agent senses and acts in a task environment and receives numeric reward values after each action. With this experience, the learning agent attempts to find behavioral policies that improve its expected accumulation of reward. A number of research projects have investigated the scenario in which this reward comes as feedback from a human

user rather than a function predefined by an expert (Isbell et al. 2006, Thomaz and Breazeal 2008, Knox and Stone 2012). In evaluating the feasibility of non-experts teaching through reward signals, these researchers aim to both leverage human knowledge to improve learning speed and permit people to customize an agent's behavior to fit their own needs.

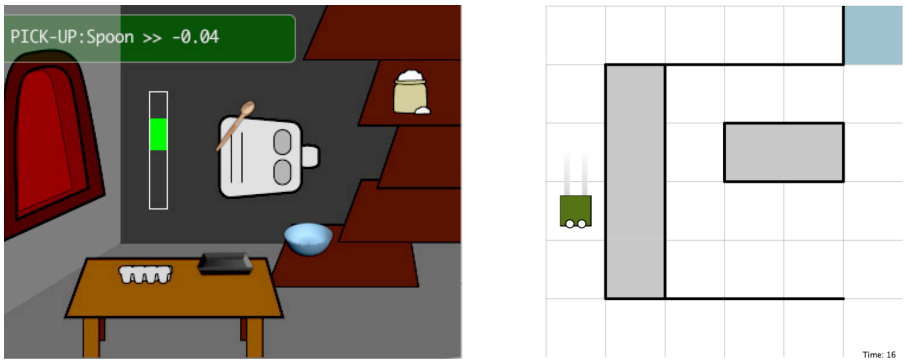

Figure 4: Task domain containing reinforcement learning agents taught by human users. (Left) A cooking robot that must pick up and use the ingredients in an acceptable order (Thomaz and Breazeal, 2006). The green vertical bar displays positive feedback given by a click-and-drag interface. (Right) A simulated robot frog that users are asked to teach to go to the water (Knox and Stone, 2012).

Thomaz and Breazeal (2008) observed that people have a strong tendency to give more positive rewards than negative rewards. Knox and Stone (2012) later confirmed this positive bias in their own experiments. They further demonstrated that such positive bias leads many agents to avoid the goal that trainers are teaching it to reach (e.g. the water in Figure 4). This undesirable consequence occurs with a common class of reinforcement learning algorithms: agents that value reward accrued over the long term and are being taught to complete so-called episodic tasks. This insight provided justification for the previously popular solution of making agents that hedonistically pursue only short-term human reward, and it led Knox and Stone to create the first reported algorithm that successfully learns by valuing human reward that can be gained in the long-term (2013). Agents trained through their novel approach were more robust to environmental changes and behaved more appropriately in unfamiliar states. These agents and the algorithmic design guidelines Knox and Stone created were the result of multiple iterations of user studies, which identified positive bias and then verified its hypothesized effects.

People want to guide and demonstrate, not just provide feedback

In the experiment by Thomaz and Breazeal (2008) users trained a simulated agent to bake a cake through a series of object manipulations. Users gave feedback to the learner by clicking and dragging a mouse. Longer drags gave larger-magnitude reward values and the drag direction determined the valence (+/-) of the reward value. Further, users could click on specific objects to signal that the feedback was specific to that object, but they were told that they could not communicate which action the agent should take.

Thomaz and Breazeal found evidence that people nonetheless gave positive feedback to objects that they wanted the agent to manipulate. These users violated their instructions by applying what could be considered an irrelevant degree of freedom—giving feedback to objects that had not been recently manipulated—to provide guidance to the agent. After Thomaz and

Breazeal adapted the agent's interface and algorithm to incorporate guidance, the agent's learning performance significantly improved.

Other researchers have reached similar conclusions. In a Wizard-of-Oz study (i.e., the agent's outputs were secretly provided by a human) by Kaochar et al. (2011), users taught an agent to perform a complex task. These users could provide a demonstration, give feedback, teach a concept by example, or test the agent to see what it had learned. The authors found that users never taught exclusively by feedback, instead generally using it *after* teaching by other means. Together, these two studies provide insight into the design of natural interfaces for teaching agents.

People may want to provide richer feedback

Labeling data remains the most popular method for end-user input to interactive machine learning systems because of its simplicity and ease-of-use. However, as some of the previous case studies demonstrate, label-based input also has drawbacks (e.g., negative attitudes towards being treated as an oracle). In addition, emerging research suggests that in some circumstances users may desire richer control over machine learning systems than simply labeling data.

For example, Stumpf et al. (2007) conducted an experiment to understand the types of input end-users might provide to machine learning systems if unrestricted by the interface. The authors generated three types of explanations for predictions from a text classification system operating over email messages. These explanations were presented to people in the form of paper-based mockups to avoid the impression of a finished system and to encourage people to provide more feedback. People then provided free-form feedback on the paper prototypes in attempts to correct the classifier's mistakes.

This experiment generated approximately 500 feedback instances from participants, which were then annotated and categorized. The authors found that people naturally provided a wide variety of input types to improve the classifier's performance, including suggesting alternative features to use, adjusting the importance or weight given to different features, and modifying the information extracted from the text. These results present an opportunity to develop new machine learning algorithms that might better support the natural feedback people want to provide to learners, rather than forcing users to interact in limited, learner-centric ways.

End users may value further transparency

In addition to wanting richer controls, people sometimes desire more transparency about how their machine learning systems work. In a recent study, Kulesza et al. (2012) provided users of a content-based music recommender with a 15-minute tutorial discussing how the recommender worked and how the various feedback controls (e.g., rating songs, steering towards specific feature values, etc.) would impact the learner. Participants responded positively to learning these details about the system. In addition, the researchers found that the more participants learned about the recommender while they interacted with it, the more satisfied they were with the recommender's output. This case study provides evidence that users do not

always want “black box” learning systems—sometimes they want to provide nuanced feedback to steer the system, and they are willing and able to learn details about the system to do so.

Transparency can help users label better

Sometimes users make mistakes while labeling, thus providing false information to the learner. Although most learning systems are robust to the occasional human error, Rosenthal and Dey set out to solve this problem at the source. They sought to reduce user mistakes by providing targeted information when a label is requested in an active learning setting. The information provided to the user included a combination of contextual features of the sample to be labeled, explanations of those features, the learner's own prediction of the label for the sample, and its uncertainty in this prediction (Rosenthal & Dey, 2010).

They conducted two studies to determine the subset of such information that is most effective in improving users' labeling accuracy. The first involved people labeling strangers’ emails into categories, as well as labeling the interruptability of strangers' activities; the second study involved people labeling sensory recordings of their own physical activity. Both studies found that the highest labeling accuracy occurred when the system provided sufficient contextual features and current predictions *without* uncertainty information. This line of research demonstrates that the way in which information is requested (e.g., with or without context) can greatly impact the quality of the response elicited from the user. The case study also shows that not all types of transparency improve the performance of an interactive machine learning system, and user studies can help determine the ideal combination of information to provide to users.

Summary

As these case studies illustrate, understanding how people *do* interact—and *want* to interact—with machine learning systems is critical to designing systems that people can use effectively. Exploring preferred interaction techniques through user studies can reveal gaps in designers’ assumptions about their end users and may suggest new algorithmic solutions. In some of the cases we reviewed, people naturally violated the assumptions of the machine learning algorithm or were unwilling to comply with them. Other cases demonstrate that user studies can lead to helpful changes to the types of input and output supported by interfaces for interactive machine learning. In general, this type of research can produce design suggestions and considerations, not only for people building user interfaces and developing the overall user experience, but for the machine learning community as well. Moving from this section’s focus on research that questions the assumptions of interactive machine learning systems—some of which are assumptions built into the interface—the following section will detail a number of projects that involve novel interfaces, each attempting to incorporate new types of input or output into the interactive machine learning cycle.

Novel Interfaces for Interactive Machine Learning

End users are often assumed to have limited time, patience, and capacity to understand machine learning. Perhaps as a consequence of such assumptions, interactive machine

learning systems have often been designed to receive only labeled examples as input and provide only predictions as output. However, as many of the case studies in the previous section showed, end users sometimes desire richer involvement in the interactive machine learning process. In addition, research on cost-benefit tradeoffs has shown that people will invest time and attention to something *if* they perceive their efforts to have greater benefits than costs (Blackwell 2002). For example, research on end-user programming has shown that end users program often (e.g., via spreadsheets, macros, mash-ups), but do so primarily to accomplish some larger goal (Blackwell 2002). Similarly, this theory suggests that people will invest time to improve their classifiers only if they view the task as more beneficial than costly/risky—i.e., when they perceive the benefits of producing an effective classifier as outweighing the costs of increased interaction. Therefore, we believe there is an opportunity to explore new interfaces that can better leverage human knowledge and capabilities, and demonstrate the value of doing so via interactive feedback.

In this section, we present case studies that explore novel interfaces for interactive machine learning systems and demonstrate the feasibility of richer interactions. Interface novelty in these cases can come from new methods for receiving input or providing output. New *input* techniques can give users more control over the learning system, allowing them to move beyond simply labeling examples. Such input techniques include methods for feature creation, reweighting of features, adjusting cost matrices, or modifying model parameters. Novel *output* techniques can make the system's state more transparent or understandable. For example, a system could group unlabeled data to help users label the most informative items, or it could communicate uncertainty about the system's predictions.

These case studies also reinforce our earlier argument that interactive machine learning systems should be evaluated with potential end-users. Such evaluations are needed both to validate that these systems perform well with real users and to gain critical insights for further improvement. Many of the novel interfaces detailed below were found to be beneficial, but as shown in the final two case studies, adding new types of input or output can sometimes lead to obstacles for the user or reduce the accuracy of the learner. Therefore, novel interfaces should be designed with care and appropriately evaluated before being deployed.

Supporting data selection with novel ways of presenting data to users

In many interactive machine learning processes, the user and machine iterate toward a shared understanding of a desired concept. In each iteration, the user typically assesses the quality of the current learner and then further guides the system with additional input. A common technique for conveying the quality of the current supervised learner is to present a person with all of the unlabeled data sorted by their predicted scores for one class (e.g., showing image-classification probabilities or all search results ranked by relevance to a query). Then, after evaluating this presentation, a person can decide how to proceed in training (e.g., deciding which additional examples to provide for input). Although straightforward, this technique inefficiently illustrates the quality of the current concept and provides the user with no guidance for best improving the learner.

Fogarty et al. (2008) investigated novel techniques for presenting unlabeled data to facilitate better training in CueFlik, an interactive machine learning system for image classification. Via a user study, the authors demonstrated that an alternative technique of presenting users with *only* the best- and worst-matching examples enabled people to train significantly better models than the standard technique of presenting the user with *all* of the data.

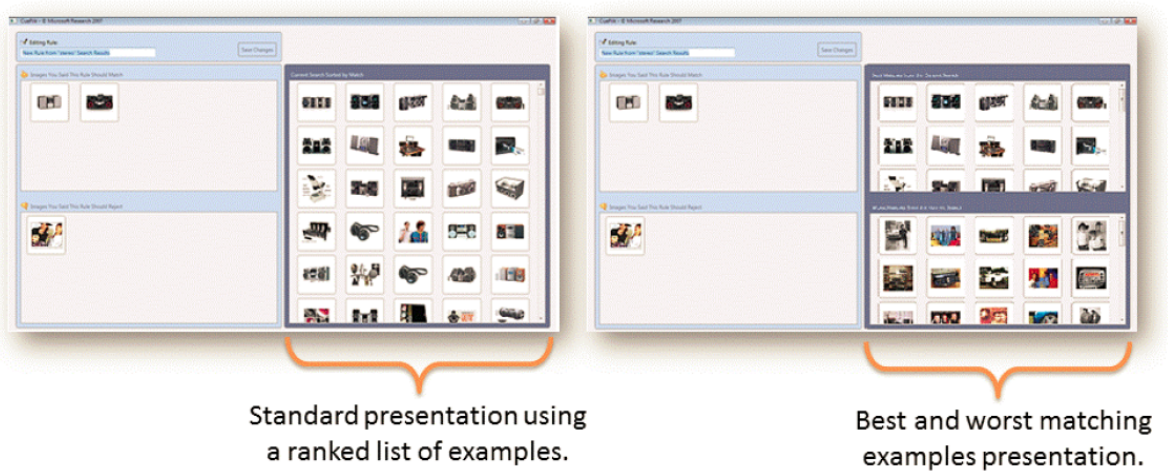

Figure 5. Fogarty et al.'s work with CueFlik compared two methods of illustrating the current version of a machine-learned visual concept. The *standard* method (left) presented users with examples ranked by their likelihood of membership to the positive class. The *best and worst matches* method (right) instead showed examples predicted as positive or negative with high certainty by CueFlik.. A user study showed that the *best- and worst-matches* technique led users to train significantly better learners than the *standard* presentation.

Fogarty et al.'s results demonstrate that presentation matters when designing end-user interaction with machine learning. They speculated that their performance improvement was due to the best-matching presentation better summarizing the machine's current understanding, helping people to focus on whether the classifier was mostly correct rather than focusing on the uncertain middle of the standard list ranked by probability of relevance. However, because best and worst matches are extremely similar to already labeled examples, this technique constrains a person to label examples that provide little additional information to the machine learner.

To address the limitations of the best-matching presentation technique, Amershi et al. (2009) explored alternative techniques for concisely summarizing the machine's current understanding while providing people with high-information-content examples to choose from during training. These techniques involved presenting users with intuitive overviews of the positive and negative regions (taking advantage of a user's ability to quickly assess similarity and variation across multiple images in a single viewing) by selecting representative examples that maximized the mutual information with the rest of the space (providing the machine learner with more information in each iteration). A follow-up user study demonstrated that these overview-based techniques led participants to train learners that performed significantly better than learners trained via the best-performing technique from previous work. This case study demonstrates

that effective interactive machine learning systems must balance the needs of both the human and the machine within their design.

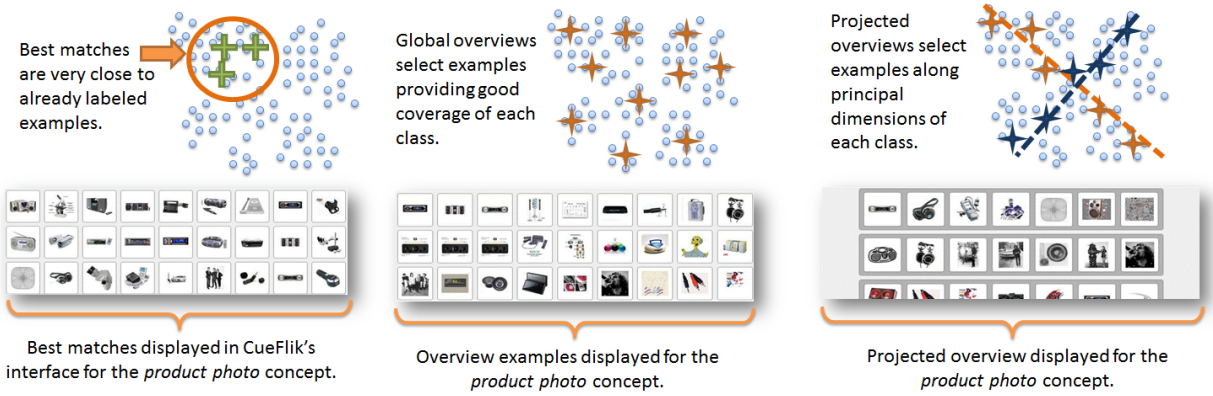

Figure 6. *Overview* presentation techniques (middle and right) more accurately illustrates CueFlik's currently learned concept while providing users with highly informative examples to choose from during interactive machine learning. Amershi et al showed that overview-based example presentation enabled end-users to train significantly better quality machine learners than the best-and-worst matches technique (which outperformed the standard presentation technique used by most interactive machine learning systems of this kind).

Intermittently-active learning: do not make queries all the time

As mentioned earlier, applying active learning to interactive settings can be undesirable from the user's point of view (e.g., users do not like to answer a constant stream of questions coming from a learning system). To address this problem, Cakmak & Thomaz (2010) proposed *intermittently-active learning*, where the learner makes queries only for a subset of the examples provided by the user. This brings a new challenge for the learner: deciding *when* to make a query. Cakmak & Thomaz explored two approaches. In the first, *conditional queries* were made only when certain conditions were met. This took into account how good the examples chosen by the user were and the probability that the user would randomly provide useful examples. In the second approach, *teacher-triggered queries* simply gave the decision of when the learner is allowed to ask a question of the teacher. A query was made only when the user said "do you have any questions?".

A study comparing intermittently-active learning with fully active and fully passive learning demonstrated its advantage over these two extremes of the spectrum (Cakmak et al. 2010). The study showed that both intermittent approaches resulted in learning as fast as the fully active approach, while being subjectively preferred over fully active or fully passive approaches. The interactions with the intermittently-active learners were found to be more balanced, enjoyable, and less frustrating. When asked to choose between the two alternative approaches, users preferred the teacher-triggered queries, mentioning that they liked having full control over the learner's queries. As exemplified in this case study, building interactive learning systems that fit user preferences can sometimes require the modification of existing methods in fundamental ways.

User feedback on system recommendations

Some machine learning systems help users navigate an otherwise unnavigable search space. For example, recommender systems help people find specific items of interest, filtering out irrelevant items. Vig et al. (2011) studied a common problem in this domain—recommending results that are close, but not quite close enough, to what the user was looking for. Researchers developed a prototype to support tag-based "critiques" of movie recommendations. Users could respond to each recommendation with refinements such as "Like this, but less violent" or "Like this, but more cerebral", where *violent* and *cerebral* are tags that users had applied to various movies. A *k*-nearest-neighbor approach was then used to find similar items that included the user-specified tags.

This relatively simple addition to the MovieLens website garnered an overwhelmingly positive reaction, with 89% of participants in a user study saying that they liked it, and 79% requesting that it remain a permanent feature on the site. In the words of one user, "The best thing to come by in MovieLens (besides the product itself). Strongly recommended this to my friends and some picked MovieLens up just because of this addition. Love it!"'. This example helps illustrate both the latent desire among users for better control over machine learning systems, and that by supporting such control in an interactive fashion, user attitudes toward the learner can be greatly enhanced.

Allowing users to specify preferences on errors

People sometimes need to refine the decision boundaries of their learners. In particular, for some classifiers it might be critical to detect certain classes correctly, while tolerating errors in other classes (e.g., misclassifying spam as not spam is typically less costly than misclassifying regular email as spam). However, refining classifier decision boundaries is a complex process even for experts, involving iterative parameter tweaking, retraining, and evaluation. This is particularly difficult because there are often dependencies among parameters, which leads to complex mappings between parameter values and the behavior of the system.

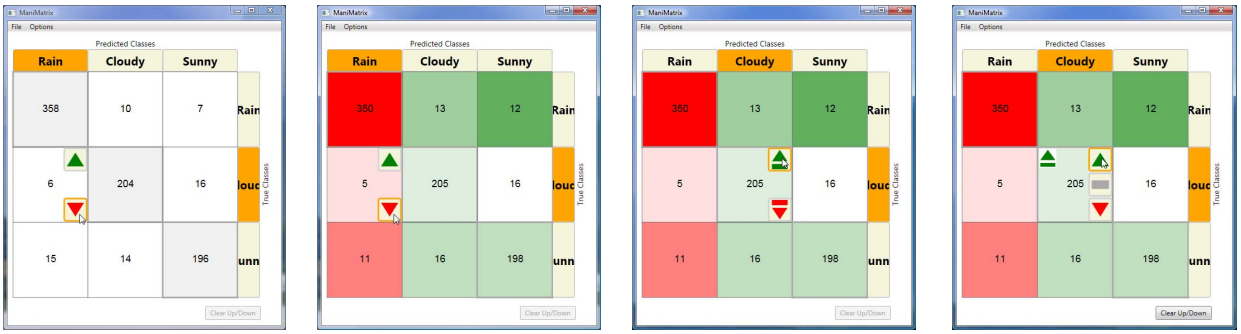

Figure 7: The ManiMatrix system displays the confusion matrix of the classifier and allows the user to directly increase or decrease the different types of errors using arrows on the matrix cells. ManiMatrix provides feedback to the user by highlighting cells that change value as a result of the user's click (red indicates a decrease and green indicates an increase).

To address these difficulties, Kapoor et al. (2010) created ManiMatrix, a tool for people to specify their preferences on decision boundaries via interactively manipulating a classifier's confusion matrix (i.e., a breakdown of the correct and incorrect predictions it made for each

class) (Figure 7). Given these preferences, ManiMatrix employs Bayesian decision theory to compute decision boundaries that minimize the expected cost of different types of errors, and then visualizes the results for further user refinement. A user study with machine learning novices demonstrated that participants were able to quickly and effectively modify decision boundaries as desired with ManiMatrix. This case study demonstrates that non-experts can directly manipulate a model's learning objective, a distinctly different form of input than choosing examples and labeling them.

Combining classifiers to improve performance

An ensemble classifier is a classifier that builds its prediction from the predictions of multiple sub-classifiers, each of which are functions over the same space as the ensemble classifier. Such ensembles often outperform all of their sub-classifiers and are a staple of applied machine learning (e.g., AdaBoost). A common workflow for creating ensemble classifiers is to experiment with different features, parameters, and algorithms via trial and error or hill-climbing through the model space. Even for machine learning experts, this approach can be inefficient and lead to suboptimal performance.

To facilitate the creation of ensemble classifiers, Talbot et al. (2009) developed EnsembleMatrix, a novel tool for helping people interactively build, evaluate, and explore different ensembles (Figure 8). EnsembleMatrix visualizes the current ensemble of individual learners via a confusion matrix. The user can then experiment with and evaluate different linear combinations of individual learners by interactively adjusting the weights of all models via a single 2D interpolation widget (top right in Figure 8). EnsembleMatrix's novel interface also allows people to make use of their visual processing capabilities to partition the confusion matrix according to its illustrated performance, effectively splitting the ensemble into sub-ensembles that can be further refined as necessary.

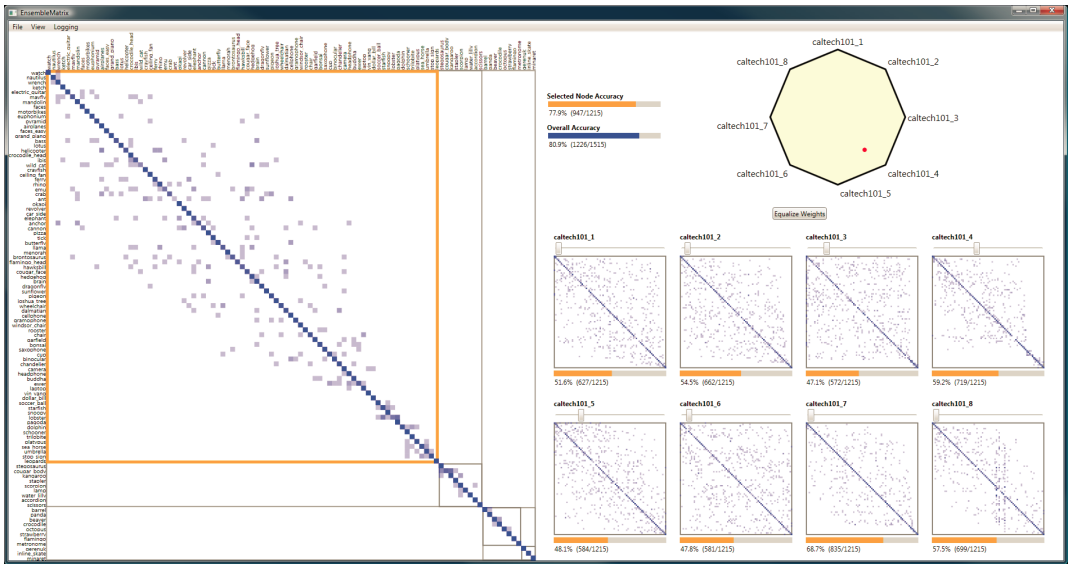

Figure 8: EnsembleMatrix visualizes the current ensemble (left) of individual learners (bottom right) via a confusion matrix. Users can adjust the weights of individual models via a linear combination widget (top

right) to experiment with different ensembles. Users can also partition the confusion matrix to split and refine sub-ensembles.

A user study showed that EnsembleMatrix enabled people to not only create ensemble classifiers on par with the best published ensembles on the same data set—they managed to do so in a single, one-hour session. The study involved participants ranging from machine learning novices to experts. This case study illustrates that effectively combining human intuition and input with machine processing can enable people to create better classifiers in less time than standard approaches that ignore these powerful human capabilities.

Allowing users to ask "Why?"

In addition to the learner querying the user (e.g., active learning), sometimes the user may want to query the learner. Kulesza et al. (2011) developed an approach to let users ask a text classifier why it was behaving in a particular way (e.g., "Why was this classified as *X* instead of *Y*?"). The learner's responses were interactive, thus providing a way for users to not only understand why the system had made a particular prediction, but to also adjust the learner's reasoning if its prediction was wrong.

While many participants exposed to this why-oriented approach significantly increased the accuracy of their naïve Bayes text classifier, every participant encountered a number of barriers while doing so. In particular, participants had trouble selecting features to modify from the thousands in the bag-of-words feature set, and once participants did select features to adjust, they had trouble understanding how changes to a single feature altered the learner's predictions for apparently unrelated items. This study suggests that for learners with large feature sets or complex interactions between features, users will need additional support to make sense of which features are most responsible for an item's classification. Conversely, these results may be interpreted as evidence that learning systems intended for interactive use must be designed such that only a comprehensible number of features are responsible for each prediction.

Summary

Whether a candidate interface change will improve a user's experience or the system's performance can only be assessed through evaluation with potential end-users. In the case studies above, adding richness or permitting user interaction with more than the training data was often beneficial, but not always so. Different users will have different needs and expectations of the systems they employ. Thus, conducting user studies of novel interactive machine learning systems is critical not only for discovering promising modes of interaction, but also to uncover obstacles that users may encounter and unspoken assumptions they might hold about the machine learner. In addition, the accumulation of such research can facilitate the development of design guidelines for building future interactive machine learning systems, much like those that exist for traditional software systems (Shneiderman et al. 2009).

Discussion

Interactive machine learning is a potentially powerful technique for improving human interaction with machine learning systems. As this article illustrates, studying how people interact with interactive machine learning systems and exploring new techniques for enabling those

interactions can result in both better user experiences and more effective machine learners. However, research in this area has only just begun, and many opportunities remain to improve the interactive machine learning process. This section describes open challenges and opportunities for advancing the state-of-the-art in human interaction with interactive machine learning systems.

As shown by the variety of case studies above, various fields of computer science already employ interactive machine learning to solve domain specific problems (e.g., search in information retrieval, filtering in recommender systems, task learning in human-robot interaction). However, different fields often refer to interactive machine learning in domain-specific terms (e.g., relevance feedback, programming by demonstration, debugging machine-learned programs, socially-guided machine learning). This diversity in terminology impedes awareness of progress in this common space, which can potentially lead to duplicate work. Seeking to facilitate the development of new interactive machine learning systems, some researchers have begun developing design spaces that abstract away domain-specific details from existing solutions to emphasize common dimensions that can be varied or manipulated when designing the interactive machine learning loop itself (e.g., Amershi 2012, Porter et al. 2013).

For example, Amershi (2012) examined interactive machine learning systems across several fields (including information retrieval, context-aware computing, and adaptive and intelligent systems) and identified 16 key design factors influencing human interaction with machine learning systems (e.g., the expected duration of model use, the focus of a person's attention during interaction, the source and type of data over which the machine will learn) and 18 design dimensions that can be varied to address these factors (e.g. the type and visibility of model feedback, the granularity and direction of user control, and the timing and memory of model input). In another example, Porter et al. (2013) breaks down the interactive machine learning process into three dimensions: task decomposition (defining the level of coordination and division of labor between the end-user and the machine learner), training vocabulary (defining the type of input end-users can provide the machine learner), and the training dialog (defining the level and frequency of interaction between the end-user and the learner). Design spaces such as these can help to form a common language for researchers and developers to communicate new interactive machine learning solutions and share ideas. However, there are many ways to dissect and describe the various interaction points between people and machine learners within the interactive machine learning process, so an important opportunity remains for converging on and adopting a common language across these fields to help accelerate research and development in this space.

In addition to developing a common language, an opportunity remains for generalizing from existing solutions and distilling principles and guidelines for how we *should* design future human interaction with interactive machine learning, much like we have for designing traditional interfaces (e.g., Schneiderman et al. 2009; Moggridge & Smith 2007; Dix et al. 2004; Winograd, 1996; Norman, 1988). For example, Schneiderman's Golden Rules of interface design advocate for designating the users as the controllers of the system and offering them informative feedback after each interaction. Indeed, some of these principles can directly translate to the design of interactive machine learning—interactive machine learning systems inherently provide

users with feedback about their actions and, as this article discusses, giving users more control of over machine learning systems can often improve a user's experience.

However, interactive machine learning systems also often inherently violate many existing interface design principles. For example, research has shown that traditional interfaces that support understandability (e.g., systems that are predictable or clear about how they work) and actionability (e.g., systems that make it clear how a person can accomplish their goals and give them the freedom to do so) are generally more usable than systems that do not support these principles. Many machine learning algorithms, however, are inherently difficult for end-users and experts to fully understand, which can result in unpredictable behaviors (Shneiderman and Maes, 1997). As another example, the goal of giving users control becomes less defined when the system autonomously acts or makes predictions. The concept of control may not be applicable. Thus, there is an opportunity to explore how current design principles apply to the human-computer partnership in interactive machine learning. New principles and guidelines may provide critical progress.

Some researchers have started to suggest new principles for designing end-user interaction with general artificially intelligent systems, many of which could translate to end-user interaction with interactive machine learning (e.g., Norman, 1994; Höök, 2000; Horvitz, 1999; Jameson, 2009). For example, Norman (1994) and Höök (2000) both identified safety and trust as key factors to consider when designing intelligent systems, referring to the assurance against and prevention of unwanted adaptations or actions. Others have stated that artificially intelligent and machine-learning-based systems should manage expectations to avoid misleading or frustrating the end user during interaction (e.g., Norman, 1994; Höök, 2000; Jameson, 2009). In Horvitz's formative paper on mixed-initiative interfaces (1999), he proposed several principles for balancing artificial intelligence with traditional direct-manipulation constructs. For example, Horvitz emphasized consideration of the timing of interactive intelligent services, limiting the scope of adaptation or favoring direct control under severe uncertainty, and maintaining a working memory of recent interactions. While these suggestions can help guide the design of future systems, more work remains to develop a comprehensive set of guidelines and principles that work in various settings. Often such design principles are distilled from years of experience developing for such interactions. Alternatively, we may accelerate the development of such guidelines by extracting dimensions that can be manipulated to design interactive machine learning systems and systematically evaluating general solutions in varying settings.

Although such systematic evaluation can facilitate generalization and transfer of ideas across fields, the interleaving of human interaction and machine learning algorithms makes reductive study of design elements difficult. For example, it is often difficult to tease apart whether failures of proposed solutions are due to limitations of the particular interface or interaction strategies used, the particular algorithm chosen, or the combination of the interaction strategy with the particular algorithm used. Likewise, inappropriately attributing success or failure to individual attributes of interactive machine learning solutions can be misleading. Therefore, new evaluation techniques may be necessary to appropriately gauge the effectiveness of new interactive machine learning systems.

Most of the case studies in this article focused on a single end-user interacting with a machine learning system. However, the increasing proliferation of networked communities and crowd-powered systems provides evidence of the power of the masses to collaborate and produce content. An important opportunity exists to investigate how crowds of people might collaboratively drive interactive machine learning systems, potentially scaling up the impact of such systems. For example, as interactive machine learning becomes more prevalent in our everyday applications, people should be able to share and re-use machine learners rather than starting from scratch. Moreover, people should be able to bootstrap, build upon, and combine learners to configure more sophisticated data processing and manipulation. A few have started to explore such opportunities (e.g., Hoffman et al. 2009; Kamar et al. 2012; Law and von Ahn 2009), but more work remains to fully understand the potential of multiple end-users interacting with machine learning systems. For example, work remains in understanding how people can meaningfully describe, compare, and search for existing machine learners in order to build upon them, in understanding how learners can be generalized or transformed for new situations and purposes, in understanding how we can create composable learners to enable more powerful automation, and in understanding how we can coordinate the efforts of multiple people interacting with machine learning systems.

Finally, the inherent coupling of the human and machine in these systems underscores the need for collaboration across the fields of human-computer interaction and machine learning. For example, as some of the case studies described in this article showed, users may desire to interact with machine learning systems in ways unanticipated by the developers of those systems. This presents an opportunity to develop new machine learning algorithms to support natural user interactions. When dealing with noisy systems, for example, machine learning researchers have often attempted to develop algorithms that work despite the noise, whereas human-computer interaction researchers often try to develop interaction techniques to reduce the noise end-users provide. Collaboration between these two communities could leverage the benefits of both solutions.

Conclusion

The case studies described in this paper support three key points. First, *interactive machine learning differs from traditional machine learning.* Interactivity creates a loop whereby the output of the learning system impacts the subsequent input from the user. This interactivity creates a partnership in which both the end user and the machine learner can learn from one another.

Second, *explicitly studying the users of learning systems is important*. Formative user studies can inspire new ways in which users could interact with learning systems and characterize user needs and desires. User studies that evaluate interactive learning systems can reveal false assumptions about potential users and common patterns in their interaction with the system. They also allow identifying difficulties commonly faced by users when novel interfaces are introduced.

Finally, *the interaction between learning systems and their users need not be limited.* We can build powerful interactive machine learning systems by giving more control to end-users than the ability to label instances and by providing users with more transparency than just the

learner's predicted outputs. However, more control for the user and more transparency from the learner do not automatically result in better systems—we must continue to evaluate novel interaction methods with real users to understand whether they help or hinder users' goals.

In addition to demonstrating the importance and potential of research in interactive machine learning, we characterized some of the challenges and opportunities that currently confront this field. By acknowledging and embracing these challenges, we can move the field of interactive machine learning forward toward more beneficial human-computer partnerships. Such partnerships, we believe, will lead to not only more capable machine learners, but more capable end-users as well.

Acknowledgements

This work was partially supported by XX and YY.

References

Amershi, S. 2012. Designing for Effective End-User Interaction with Machine Learning. Ph.D. Dissertation. University of Washington, Seattle, WA.

Amershi, S., Cakmak, M., Knox, W. B., Kulesza, T., & Lau, T. 2013. IUI workshop on interactive machine learning. In *Proceedings of the 2013 International Conference on Intelligent User Interfaces companion* (pp. 121-124). ACM.

Amershi, S., Fogarty, J., Kapoor., A. and Tan, D. 2009. Overview-Based Example Selection in Mixed-Initiative Concept Learning. In *Proceedings of the ACM Symposium on User Interface Software and Technology, 2009 (UIST 2009)*, pp. 247-256.

Blackwell, A. F. 2002. First steps in programming: A rationale for attention investment models. In *Human Centric Computing Languages and Environments, 2002. Proceedings. IEEE 2002 Symposia on* (pp. 2-10). IEEE.

Cakmak, M., Chao, C., & Thomaz, A. L. 2010. Designing interactions for robot active learners. *Autonomous Mental Development, IEEE Transactions on*, *2*(2), 108-118.

Cakmak, M., & Thomaz, A. L. 2010. Optimality of human teachers for robot learners. In *Development and Learning (ICDL), 2010 IEEE 9th International Conference on* (pp. 64-69). IEEE.

Caruana, R., Elhaway, M., Nguyen, N., & Smith, C. 2006. Meta clustering. In *Sixth IEEE International Conference on Data Mining, 2006. (ICDM'06).*(pp. 107-118)

Cohn, D., Caruana, R., & McCallum, A. 2003. Semi-supervised clustering with user feedback. *Constrained Clustering: Advances in Algorithms, Theory, and Applications*, *4*(1), 17-32.

Dix, A., Finlay, J., Abowd, G.D and Beal, R. (2004) Interaction Design Basics. *Ch. 5 in human computer interaction* (3rd ed). Harlow, England: Pearson Education Ltd, pp. 189-224.

Fails, J. A., & Olsen Jr, D. R. 2003. Interactive machine learning. In *Proceedings of the 8th international conference on Intelligent user interfaces* (pp. 39-45). ACM.

Fiebrink, R., Cook, P. R., & Trueman, D. 2011. Human model evaluation in interactive supervised learning. In Proceedings of the Conference on Human Factors in Computing Systems (CHI 2011), 147–156. ACM Press.Fogarty, J., Tan, D., Kapoor, A., & Winder, S. 2008. CueFlik: interactive concept learning in image search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 29-38). ACM.

Guillory, A., & Bilmes, J. A. 2011. Simultaneous learning and covering with adversarial noise. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)* (pp. 369-376).

Hoffman R., Amershi, S., Patel, K., Wu, F., Fogarty, J., and Weld, D.S. 2009. Amplifying Community Content Creation with Mixed-Initiative Information Extraction. . In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2009),* pp. 1849-1858.

Höök, K. 2000. Steps to take before intelligent user interfaces become real. *Interacting with computers*, *12*(4), 409-426.

Horvitz, E. 1999. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 159-166). ACM.

Isbell Jr., C. L., Kearns, M., Singh, S., Shelton, C. R., Stone, P., & Kormann, D. 2006. Cobot in LambdaMOO: An adaptive social statistics agent. *Autonomous Agents and Multi-Agent Systems*, *13*(3), 327-354.

Jameson, A. 2009. Adaptive interfaces and agents. *Human-Computer Interaction: Design Issues, Solutions, and Applications*, *105*.

Kaochar, T., Peralta, R. T., Morrison, C. T., Fasel, I. R., Walsh, T. J., & Cohen, P. R. 2011. Towards understanding how humans teach robots. In *User modeling, adaption and personalization* (pp. 347-352). Springer Berlin Heidelberg.

Kamar, E., Hacker, S., & Horvitz, E. 2012. Combining Human and Machine Intelligence in Large-scale Crowdsourcing. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012).*

Kapoor, A., Lee, B., Tan, D., & Horvitz, E. 2010. Interactive optimization for steering machine classification. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1343-1352). ACM.

Knox, W. B., & Stone, P. 2012. Reinforcement learning from human reward: Discounting in episodic tasks. In *RO-MAN, 2012 IEEE* (pp. 878-885). IEEE.

Knox, W. B., & Stone, P. 2013. Learning non-myopically from human-generated reward. In *Proceedings of the 2013 International Conference on Intelligent User Interfaces* (pp. 191-202). ACM.

Kulesza, T., Stumpf, S., Wong, W. K., Burnett, M. M., Perona, S., Ko, A., & Oberst, I. 2011. Why-oriented end-user debugging of naive Bayes text classification. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, *1*(1), 2.

Kulesza, T., Stumpf, S., Burnett, M., & Kwan, I. 2012. Tell me more?: the effects of mental model soundness on personalizing an intelligent agent. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1-10). ACM.

Law, E. & von Ahn, R. 2009. Input-agreement: A New Mechanism for Data Collection Using Human Computation Games. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2009).*

Moggridge, B., & Smith, G. C. 2007. *Designing interactions* (Vol. 17). Cambridge: MIT press.

Norman, D. A. 1988. *The Design of Everyday Things*. New York: Basic books.

Norman, D. A. 1994. How might people interact with agents. *Communications of the ACM*, *37*(7), 68-71.

Porter, R., Theiler, J., & Hush, D. 2013. *Interactive Machine Learning in Data Exploitation*. Technical Report. Los Alamos National Lab.

Rosenthal, S. L., & Dey, A. K. 2010. Towards maximizing the accuracy of human-labeled sensor data. In *Proceedings of the 15th international conference on Intelligent user interfaces* (pp. 259-268). ACM.

Settles, B. 2010. Active learning literature survey. *University of Wisconsin, Madison*.

Shneiderman, B., & Maes, P. 1997. Direct manipulation vs. interface agents. *Interactions*, *4*(6), 42-61.

Shneiderman, B., Plaisant, C., Cohen, M., & Jacobs, S. 2009. *Designing the User Interface: Strategies for Effective Human-Computer Interaction, 5th Edition*. Addison-Wesley.

Stumpf, S., Rajaram, V., Li, L., Burnett, M., Dietterich, T., Sullivan, E., Drummond, R., & Herlocker, J. 2007. Toward harnessing user feedback for machine learning. In *Proceedings of the 12th international conference on Intelligent user interfaces* (pp. 82-91). ACM.

Talbot, J., Lee, B., Kapoor, A., & Tan, D. S. 2009. EnsembleMatrix: interactive visualization to support machine learning with multiple classifiers. In *Proceedings of the 27th international conference on Human factors in computing systems* (pp. 1283-1292). ACM.

Thomaz, A. L., & Breazeal, C. 2008. Teachable robots: Understanding human teaching behavior to build more effective robot learners. *Artificial Intelligence*, *172*(6), 716-737.

Vig, J., Sen, S., & Riedl, J. 2011. Navigating the tag genome. In *Proceedings of the 16th international conference on Intelligent user interfaces* (pp. 93-102). ACM.

Winograd, T. 1996. *Bringing Design to Software*. ACM Press.