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

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

Abstract

Systems that can learn interactively from their end-users are quickly becoming widespread. Until recently, this progress has been fueled mostly by advances in machine learning; however, more and more researchers 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 demonstrate how interactivity results in a tight coupling between the system and the user, exemplify ways in which some existing systems fail to account for the user, and explore new ways for learning systems to interact with their users. After giving a glimpse of the progress that has been made thus far, we discuss some of the challenges we face in moving the field forward.

Introduction

Machine learning is a powerful tool for transforming data into computational models that can drive user-facing applications. However, potential users of such applications, who are often domain experts for the application, have limited involvement in the process of developing them. The intricacies of applying machine learning techniques to everyday problems has largely restricted their use to skilled practitioners. In the traditional applied machine learning workflow, these practitioners collect data, select features to represent the data, pre-process and transform the data, choose a representation and learning algorithm to construct the model, tune parameters of the algorithm, and finally assess the quality of the resulting model. This assessment often leads to further iterations on many of the previous steps. Typically, any end-user involvement in this process is mediated by the practitioners and is limited to providing data, answering domain-related questions, or giving feedback about the learned model. This results in a design process with lengthy and asynchronous iterations and limits the end-users' ability to impact the resulting models.

Consider the following case study of machine-learning practitioners working with biochemists to develop a protein taxonomy by clustering low-level protein structures (Caruana et al. 2006). The project lead recounted their experience in an invited talk at the IUI 2013 Workshop on Interactive Machine Learning (Amershi et al. 2013). First, the practitioners would create a clustering of the protein structures. Then, they would meet with the biochemists to discuss the results. The biochemists would critique the results (e.g., "these two proteins should / should not be in the same cluster" or "this cluster is too small"), providing new constraints for the next

[1] All authors contributed equally.

iteration. Following each meeting, the practitioners would carefully adjust the clustering parameters to adhere to the given constraints and re-compute clusters for the next meeting. Frustrated by the inefficiency of this laborious process, Caruana *et al*. went on to develop learning algorithms that enable *interactive* exploration of the clustering space and incorporation of new clustering constraints (Cohn et al. 2003, Caruana et al. 2006). These algorithms were intended to give people the ability to rapidly iterate and inspect many alternative clusterings within a single sitting.

Their later approach is an example of *interactive machine learning*, where learning cycles involve more *rapid*, *focused*, and *incremental* model updates than in the traditional machine learning process. These properties enable everyday users to interactively explore the model space through trial-and-error and drive the system towards an intended behavior, reducing the need for supervision by practitioners. Consequently, interactive machine learning can facilitate the democratization of applied machine learning, empowering end-users to create machine learning-based systems for their own needs and purposes. However, enabling effective end-user interaction with interactive machine learning introduces new challenges that require a better understanding of end-user capabilities, behaviors, and needs.

This article promotes the empirical study of the *users* of interactive machine learning systems as a method for addressing this challenge. Through a series of case studies, we illustrate the following propositions:

1) Rapid, focused and incremental learning cycles result in a tight coupling between the user and the system, where the two influence one another. As a result it is difficult to decouple their influence on the resulting model and study such systems in isolation.

2) Explicitly studying user interaction can challenge assumptions of traditional learning systems about users and better inform the design of interactive learning systems.

3) The ways in which end-users interact with learning systems can be expanded to ways in which practitioners do (e.g., tuning parameters or defining new constraints); however, novel interaction techniques should be carefully evaluated with potential end-users.

While the presented case studies paint a broad picture of recent research in user interaction with interactive machine learning, this article does not exhaustively survey the literature in this space. Rather, these case studies are selected to highlight the role and importance of the user within the interactive machine learning process, serving as an introduction to the topic and a vehicle for considering this body of research altogether. We conclude this article with a discussion of the current state of the field, identifying opportunities and open challenges for future interactive machine learning research.

Interactive Machine Learning

The applied machine learning workflow often involves long and complex iterations. The process starts with data provided by domain experts or specifically collected for the target application. Machine learning practitioners then work with domain experts to identify features to represent the data. Next, the practitioners experiment with different machine learning algorithms,

iteratively tuning parameters, tweaking features, and sometimes collecting more data to improve target performance metrics. Results are then further examined both by practitioners and domain experts to inform the subsequent iteration. At the end of this long cycle, the model is updated in several ways and can be drastically different from the previous iteration. Furthermore, this iterative exploration of the model space is primarily driven by the machine learning practitioners, who rely on their understanding of machine learning techniques to make informed model updates in each iteration.

In contrast, model updates in *interactive machine learning* are more *rapid* (the model gets updated immediately in response to user input), *focused* (only a particular aspect of the model is updated), and *incremental* (the magnitude of the update is small; the model does not change drastically with a single update). This allows users to interactively examine the impact of their actions and adapt subsequent inputs to obtain desired behaviors. As a result of these rapid interaction cycles, even users with little or no machine learning expertise can steer machine-learning behaviors via low-cost trial-and-error or focused experimentation with inputs and outputs. Figure 1 illustrates traditional applied machine learning and interactive machine learning, highlighting their contrasting characteristics.

Perhaps the most familiar examples of interactive machine learning in real-world applications

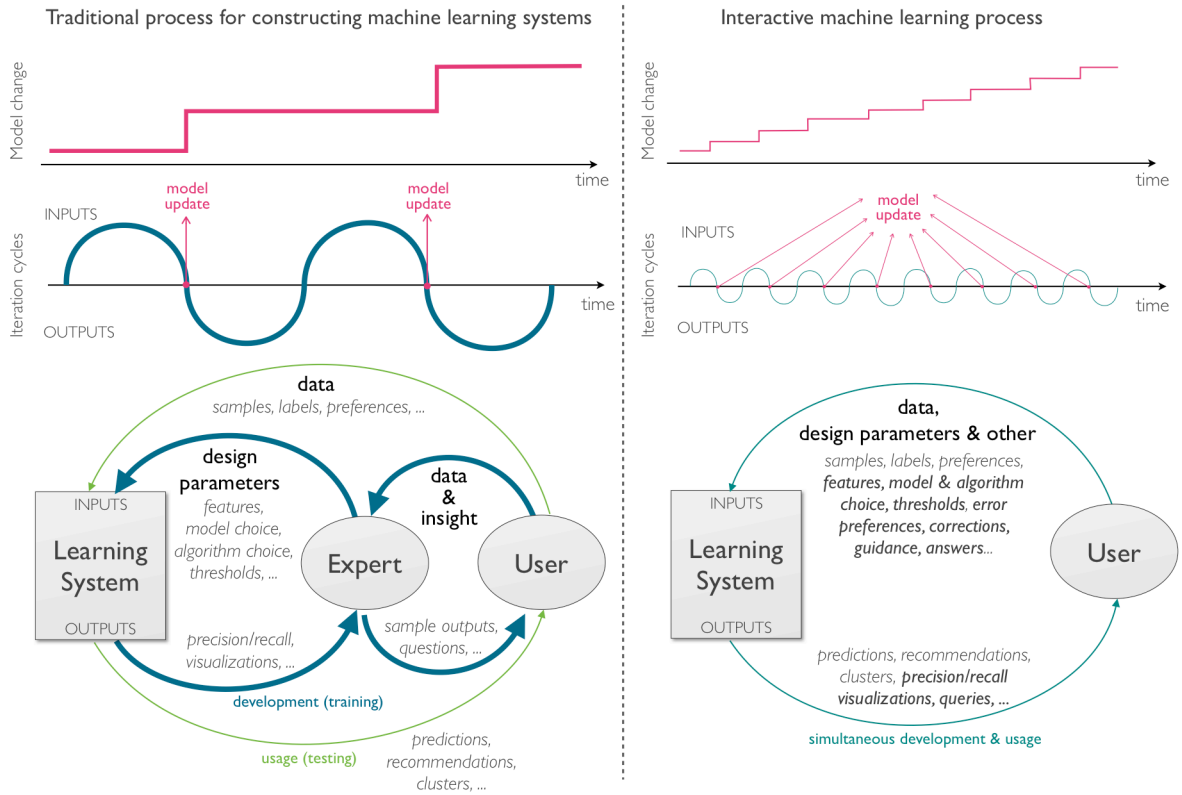

Figure 1: In machine learning, people iteratively supply information to a learning system and then observe and interpret the outputs of the system to inform subsequent iterations. In interactive machine learning, these iterations are more focused, frequent and incremental than traditional machine learning. The tighter interaction between users and learning systems in interactive machine learning necessitates an increased focus on studying the user's involvement in the process.

are *recommender systems* such as Amazon product recommendations, Netflix movie recommendations, and Pandora music recommendations. Users of recommender systems are often asked targeted questions about their preferences for individual items[2] (which they provide by 'liking' or 'disliking' them, for example). These preferences are then promptly incorporated in the underlying learning system for subsequent recommendations. If a recommender system begins recommending undesired items after incorporating new preferences, the user may attempt to redirect the system by correcting it or providing different preference information in the future.

We next present two case studies that exemplify the interactive machine learning process and demonstrate its potential as an end-user tool.

Interactive machine learning for image segmentation

Fails and Olsen (2003) were the first to introduce the term *interactive machine learning* in the human-computer interaction community, characterizing it with rapid *train-feedback-correct* cycles, where users iteratively provide corrective feedback to a learner after viewing its output. They demonstrated this process with their *Crayons* system, which allowed users with no machine learning background to train pixel classifiers 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.

Evaluations of Crayons via user studies revealed that the immediate output provided by the system allowed users to quickly view and correct misclassifications by adding new training data in the most problematic areas. As illustrated in Figure 2, after an initial classification, the user

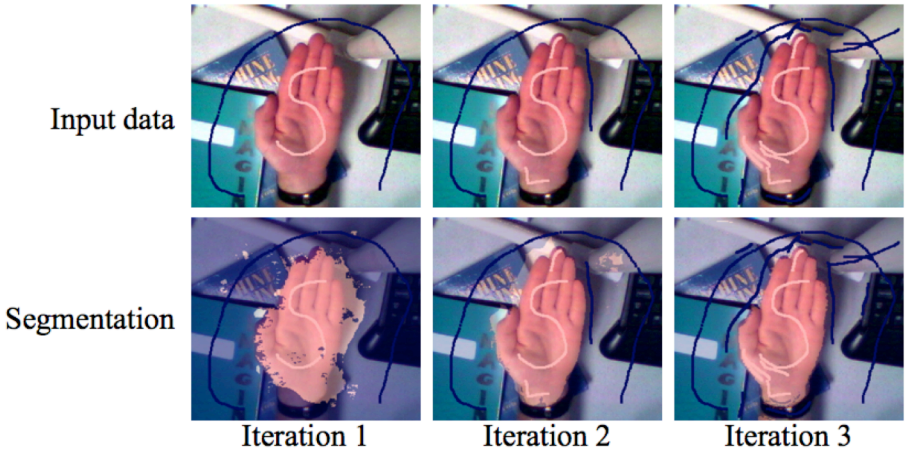

Figure 2: Interactive training of the Crayons system (Fails & Olsen 2003). The system takes pixels labeled as background/foreground as input (provided through brush strokes), and gives a fully segmented image as output (obtained through a classifier that labels each pixel as foreground/background). The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

[2] In this article we examine interactive machine learning systems in which the human is *consciously* interacting with the machine learner in order to improve it. That is, we do not consider interactive machine learning systems that obtain user feedback implicitly (e.g., websites that may automatically adapt their presentation to a user's click history without their knowledge).

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.

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.

Interactive machine learning for gesture-based music

Another example of an interactive machine learning system comes from the realm of music composition and performance. This domain is naturally interactive: musicians are accustomed to receiving immediate 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, such as moving an arm in front of a web camera to 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 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 observing students using Wekinator in an interdisciplinary music and computer science course, the authors found that as students trained their respective instruments, the interactive nature of the system also helped train the students. For example, the students learned how to recognize noise in their training samples and provide clearer examples to the learner. In some cases, students 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 flaws in her playing technique while trying to train a gesture recognizer. The process revealed that the cellist's bowing articulation was not as precise as she had believed. By observing the outputs of the system in real-time, Wekinator users were able to modify their behavior in ways that allowed them to create instruments to their satisfaction.

Summary

These examples illustrate the rapid, focused, and incremental interaction cycles fundamental to interactive machine learning; it is these cycles that facilitate end-user involvement in the machine learning process. These cycles also result in a tight coupling between user and the system, making it impossible to study the system in isolation from the user. This necessitates an increased focus on studying how users can effectively influence the machine learning system and how the learning system can appropriately influence the users. The following section examines how explicitly studying end-users can challenge assumptions of traditional machine learning and better inform the development of interactive machine learning systems. Many of the case studies to follow additionally consider less traditional types of input and output, moving beyond labeled examples and observations of learner predictions.

Studying User Interaction with Interactive Machine Learning

The increased interaction between users and learning systems in interactive machine learning necessitates an increased understanding of *how* end-user involvement impacts the learning

process. In this section, we present case studies illustrating how such an understanding can ultimately lead to better-informed system designs. First, we present case studies demonstrating how people may violate assumptions made by traditional machine learners, resulting in unexpected outcomes and user frustration. Next, we present case studies indicating that people may want to interact with machine learning systems in richer ways than anticipated, suggesting new *input* and *output* capabilities. Finally, we present case studies that experiment with increasing transparency about how machine learning systems work, finding that such transparency can improve the user experience in some scenarios, as well as the accuracy of resulting models.

Users are people, not oracles

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., samples for which the learner has maximum uncertainty). For each selected sample the learner queries an oracle to request a label. This

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: particular 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.

method has had success in accelerating learning (i.e., requiring fewer labels to reach a target accuracy) in applications like text classification and object recognition, where oracles are often paid to provide labels over a long period of time. However, Cakmak and colleagues (2010) discovered that when applied to interactive settings, such as a person teaching a task to a robot by example, active learning can cause several problems.

Cakmak's study (Figure 3) found that the constant stream of questions from the robot during the interaction was perceived 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). Guillory and Bilmes (2011) reported similar findings for an active movie recommendation system they developed for Netflix. These studies reveal that users 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 tend to give more positive than negative feedback to learners

In reinforcement learning, an 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-expert users teaching through

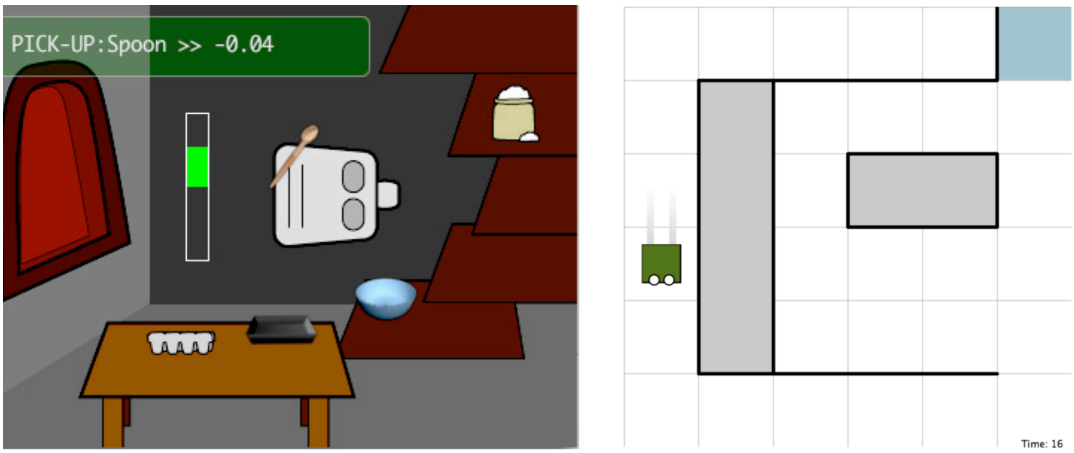

Figure 4: Two task domains for reinforcement learning agents taught by human users. (Left) A cooking robot that must pick up and use the ingredients in an acceptable order (Thomaz and Breazeal, 2006). The green vertical bar displays positive feedback given by a click-and-drag interface. (Right) A simulated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

reward signals, these researchers aimed to both leverage human knowledge to improve learning speed and permit users to customize an agent's behavior to fit their own needs.

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 bias leads many agents to avoid the goal that users 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 an 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 than did more hedonistic (i.e., myopic) variants. 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 demonstrate how learners *should* behave

In an experiment by Thomaz and Breazeal (2008) users trained a simulated agent to bake a cake through a reinforcement learning framework. In their interface, 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 (Figure 4). 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, such as an empty bowl which the agent is in position to pick up. These users violated the 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 about future actions, rather than actual *feedback* about previous actions. After Thomaz and Breazeal adapted the agent's interface and algorithm to incorporate such 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 a simulated unmanned aerial vehicle (UAV) to conduct various missions. At any time, these users chose whether to teach by demonstration, by feedback, or by providing an example of a concept. They could also 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 the other available means. Together, these two studies provide insight into the design of natural interfaces for teaching agents.

People naturally want to provide more than just data labels

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 demonstrated in previous case studies, label-based input can have drawbacks (e.g., negative attitudes towards being treated as an oracle). In addition, emerging research suggests that in some scenarios users may desire richer control over machine learning systems than simply labeling data.

For example, Stumpf et al. (2007) conducted a study 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 encourage people to provide more feedback. People were then asked to give free-form feedback on the paper prototypes with the goal of trying 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.

People value transparency in learning systems

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

Examining transparency at a more social level, Rashid et al. (2006) examined the effect of showing users the value of their potential movie ratings to a broader community in the MovieLens recommendation system. Users who were given information about the value of their contribution to the entire MovieLens community provided more ratings than those who were not given such information, and those given information about value to a group of users with similar tastes gave more ratings than those given information regarding the full MovieLens community.

Transparency can help people provide better labels

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 the labeling accuracy of users. The first involved people labeling strangers' emails into categories, as well as labeling the interruptability of strangers' activities; the second 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 presented (e.g., with or without context) can greatly impact the quality of the response elicited from the user. This case study also shows that not all types of transparency improve the performance of interactive machine learning systems, and user studies can help determine what information is most helpful to the intended audience.

Summary

Understanding how people *actually* interact—and *want* to interact—with machine learning systems is critical to designing systems that people can use effectively. Exploring interaction

techniques through user studies can reveal gaps in a designer's assumptions about their end-users and may suggest new algorithmic solutions. In some of the cases we reviewed, people naturally violated assumptions of the machine learning algorithm or were unwilling to comply with them. Other cases demonstrated that user studies can lead to helpful insights about the types of input and output that interfaces for interactive machine learning should support. 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.

Novel Interfaces for Interactive Machine Learning

As many of the case studies in the previous section showed, end-users often desire richer involvement in the interactive machine learning process than labeling instances. In addition, research on cost-benefit tradeoffs in human-computer interaction has shown that people will invest time and attention into complex tasks *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, or mash-ups), but do so primarily to accomplish some larger goal (Blackwell 2002). The act of programming is an investment, and the expected benefit is using the program to accomplish their goal sooner or with less effort than doing it manually. Similarly, this theory suggests that people will invest time to improve their machine learners only if they view the task as more beneficial than costly or risky—i.e., when they perceive the benefits of producing an effective learner as outweighing the costs of increased interaction. Therefore, we believe there is an opportunity to explore new, richer interfaces that can leverage human knowledge and capabilities more efficiently and effectively.

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 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 proposition 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 insights for further improvement. Many of the novel interfaces detailed in this section were found to be beneficial, but some of the case studies also demonstrate that certain types of input or output may lead to obstacles for the user or reduce the accuracy of the resulting learner. Therefore, novel interfaces should be designed with care and appropriately evaluated before deployment.

Supporting assessment of model quality

In each iteration of the interactive machine learning process, the user may assess the quality of the current model and then decide how to proceed with further input. A common technique for

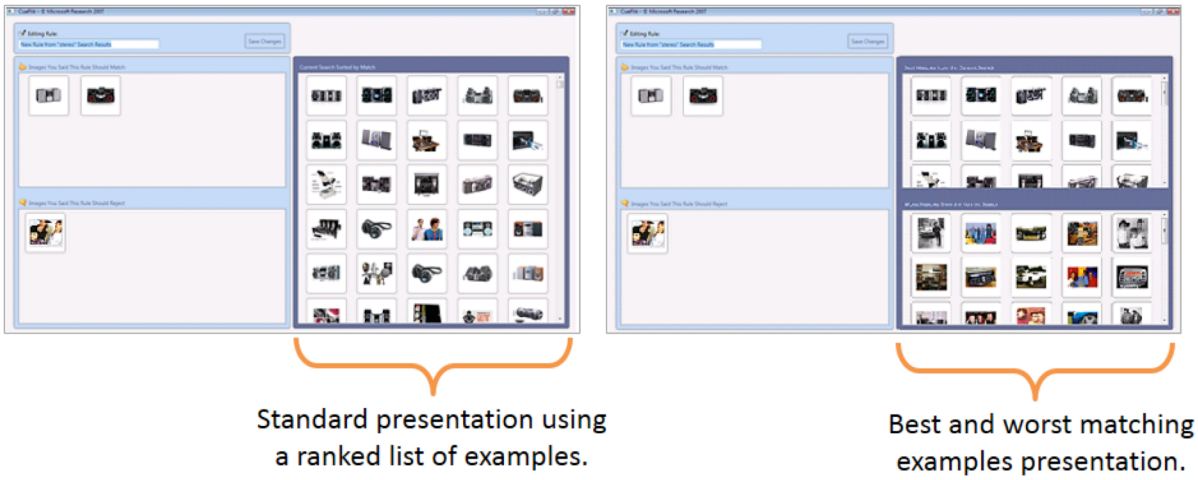

Figure 5. Fogarty et al.'s work with CueFlik compared two methods of illustrating the quality of a machine-learned visual concept. The standard method (left) presented users with examples ranked by their likelihood of membership to the positive class. The best and worst matches method (right) instead 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.

conveying model quality in supervised learning is to present a person with all of the unlabeled data sorted by their predicted scores for some class (e.g., classification probabilities or relevance rankings). After evaluating this presentation, a person then decides how to proceed in training by selecting additional examples to label for further input. Although straightforward, this technique inefficiently illustrates learner quality and provides the user with no guidance in selecting additional training examples.

Fogarty et al. (2008) investigated novel techniques for presenting model quality in CueFlik, an interactive machine learning system for image classification. Via a user study, the authors demonstrated that a technique of presenting users with *only* the best- and worst-matching examples enabled users to more quickly evaluate model quality and, in turn, train significantly better models than the standard technique of presenting the user with *all* of the data. In a follow up investigation with CueFlik, Amershi et al. (2009) went on to show that presentation techniques designed to summarize model quality for users while providing them with high-value examples to choose from as further input to the model led users to train better models than the best- and worst-matching technique from previous work. These case studies demonstrate that presentation matters and designing interfaces that balance the needs of both end-users and machine learners is more effective than optimizing user interfaces for end-users in isolation.

Supporting experimentation with model inputs

Interactive machine learning enables rapid and incremental iterations between the end-user and the machine learner. As a result, users may want to *experiment* with alternative inputs and examine resulting model outputs before committing to any model input. To support end-user experimentation, Amershi et al (2010) augmented the CueFlik system discussed previously with a history visualization to facilitate model comparison and support for model revision (via

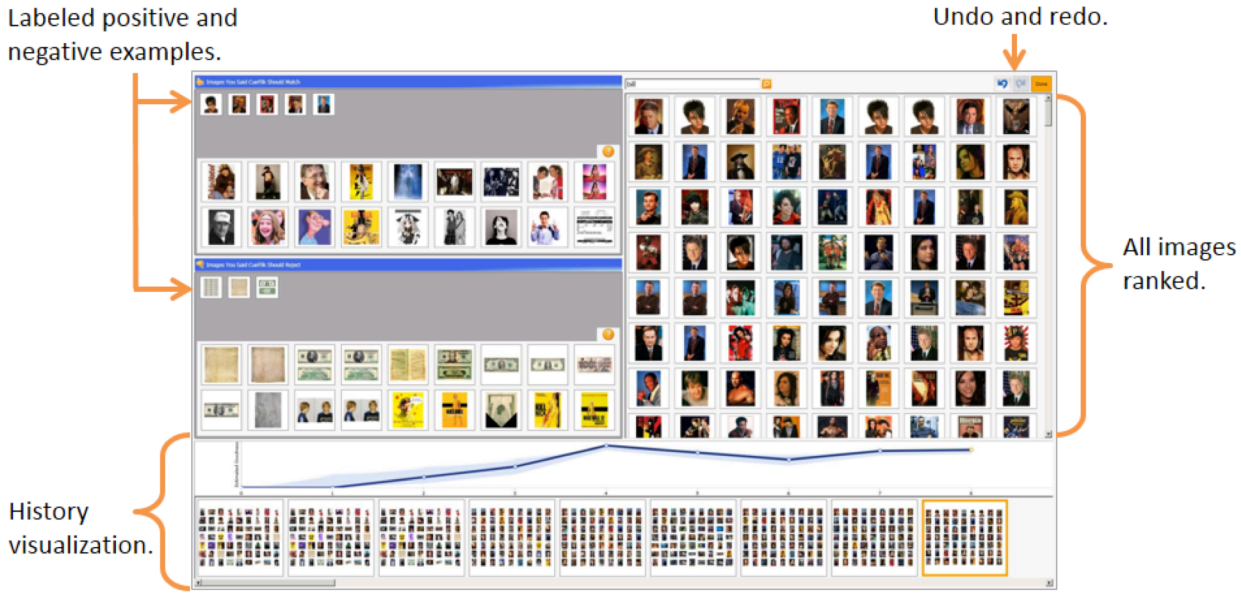

Figure 6. CueFlik augmented with a history visualization to facilitate model comparison and support for model revision. Amershi et al showed that these supports for experimentation during interactive machine learning enabled end-users to train better quality models than when these supports were unavailable.

undo/redo, removing labels, and reverting back to previous models using the history visualization). In a user study, Amershi et al. showed that end-users used revision when it was available and this led them to achieve better final models in the same amount of time (even while performing more actions) compared to when these supports were unavailable. Furthermore, being able to examine and revise actions is consistent with how people expect to interact with their applications. One participant in this study commented that without revision "it felt a little like typing on a keyboard without a backspace key." This case study illustrates that end-users may be willing and may expect options to experiment and revise their inputs to machine learners during the interactive machine learning process.

Appropriately timing queries to the user

As discussed earlier, applying active learning to interactive settings can be undesirable to the user when questions come in a constant stream from the learning system. To address this problem, Cakmak & Thomaz (2010) proposed *intermittently-active learning*, in which only a subset of the examples provided by the user are obtained through queries. This brings a new challenge for the learner: deciding *when* to query as opposed to letting the user choose an example. Cakmak & Thomaz explored two approaches. In the first, the learner made queries only when certain conditions were met. It took into account the quality of examples chosen by the user and the probability that the user could randomly provide useful examples. In the second approach, the *user* decided when the learner was allowed to ask questions (i.e., a query was made only when the user said "do you have any questions?").

A study comparing intermittently-active learning with fully active and passive learning demonstrated its advantage over these two extremes (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.

Enabling users to query the learner

In addition to the learner querying the user as in the active learning paradigm, 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 also adjust the learner's reasoning if its prediction was wrong. For example, the learner could display a bar graph showing that it associated the word "job" with the topic of "news" more than the topic of "résumés"; if the user disagreed with this reasoning, he or she could adjust the graph to tell the learner that "jobs" should be associated with "résumés" more than "news".

Most participants exposed to this *why*-oriented interaction approach significantly increased the accuracy of their naïve Bayes text classifiers; however, 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. Also, once participants did select features to adjust, they had trouble understanding how changes to a single feature altered the learner's predictions for seemingly 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.

Enabling users to critique learner output

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. 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.

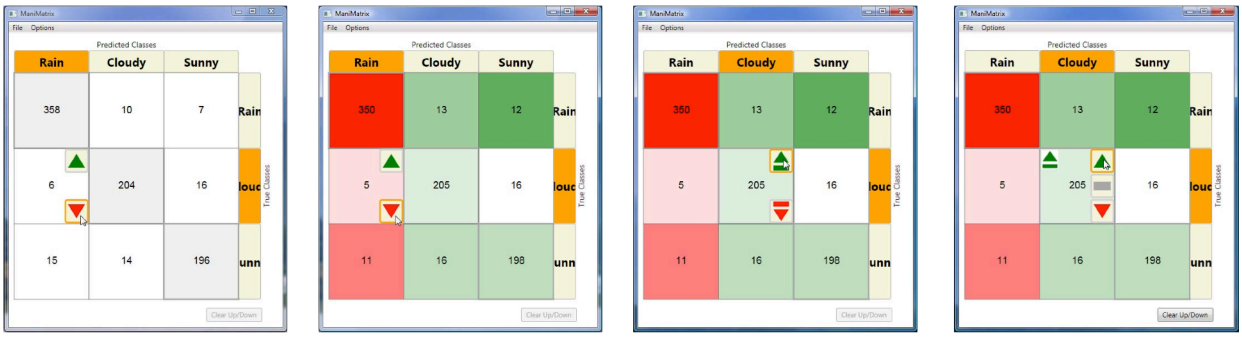

Figure 7: The ManiMatrix system displays the confusion matrix of the classifier and allows the user to directly increase or decrease the different types of errors using arrows on the matrix cells. ManiMatrix provides feedback to the user by highlighting cells that change value as a result of the user's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

Allowing users to specify preferences on errors

People sometimes want 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 regular email 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 lead to complex mappings between parameter values and the behavior of the system.

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). 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 the ManiMatrix system. 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 models

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 by Freund & Schapire (1995)). 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, however, 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

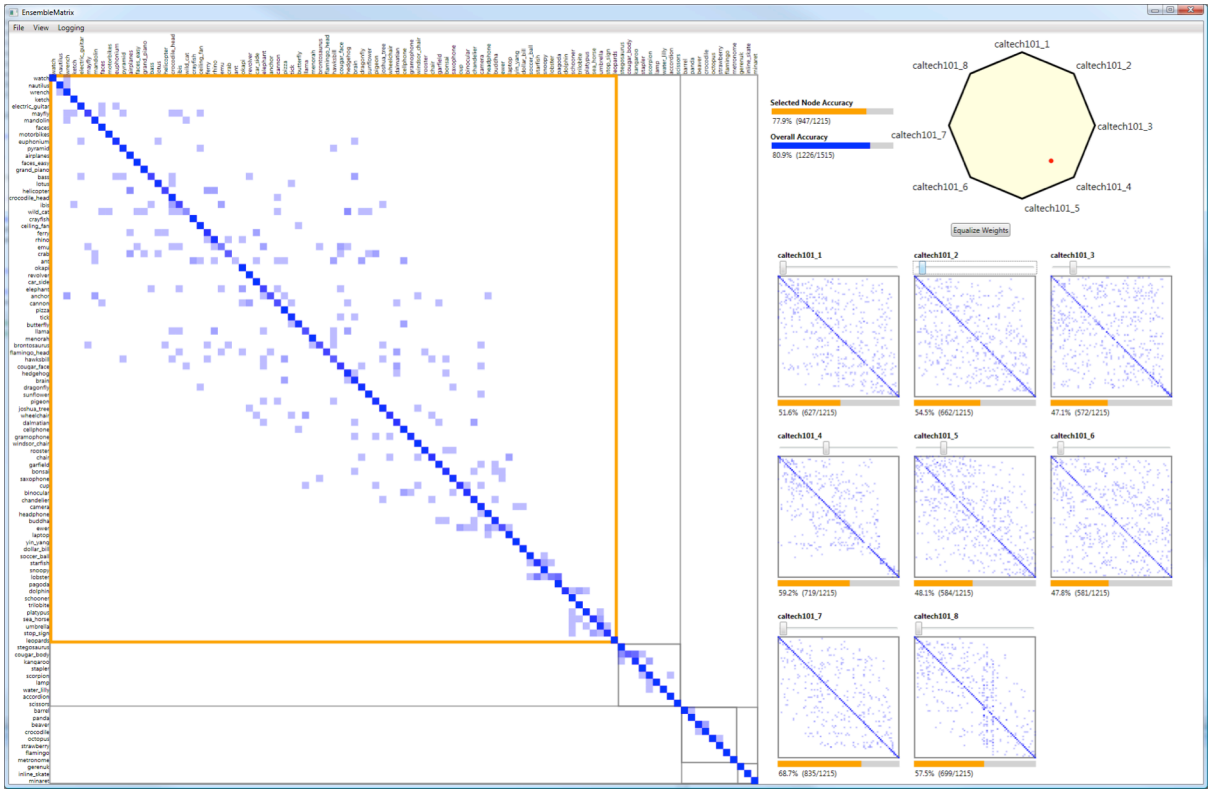

Figure 8: EnsembleMatrix visualizes the current ensemble (left) of individual learners (bottom right) via a confusion matrix. Users can adjust the weights of individual models via a linear combination widget (top right) to experiment with different ensembles. Users can also partition the confusion matrix to split and refine sub-ensembles.

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.

A user study showed that EnsembleMatrix enabled people to create ensemble classifiers on par with the best published ensembles on the same data set. Furthermore, 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.

Summary

Whether a new interface will improve the user's experience or the system's performance can only be assessed through evaluation with potential end-users. In the case studies above, permitting richer user interactions was often beneficial, but not always so. Different users have

different needs and expectations of the systems they employ. In addition, rich interaction techniques may be appropriate for some scenarios and not others. 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 in different scenarios and unspoken assumptions they might hold about machine learners. 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 (e.g., Shneiderman et al. 2009).

Discussion

Interactive machine learning is a potentially powerful technique for enabling end-user interaction with machine learning. As this article illustrates, studying how people interact with interactive machine learning systems and exploring new techniques for enabling those interactions can result in 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.

Developing a common language across diverse fields

As shown by the variety of case studies presented in this article, many fields of computer science already employ interactive machine learning to solve different problems, such as search in information retrieval, filtering in recommender systems, and task learning in human-robot interaction. However, different fields often refer to interactive machine learning or parts of the interactive machine learning process 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 and can potentially lead to duplicate work. Seeking to develop a common language and facilitate the development of new interactive machine learning systems, some researchers have begun to examine this body of work and abstract away domain-specific details from existing solutions to characterize common variables and dimensions of the interactive machine learning process 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 specific 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 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. Therefore, an important opportunity remains for converging on and adopting a common language across these fields to help accelerate research and development in this space.

Distilling principles and guidelines for *how* to design human interaction with machine learning

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.

Some principles for designing traditional interfaces 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 (i.e., systems that are predictable or clear about how they work) and actionability (i.e., 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 interfaces that do not support these principles. Many machine learning systems violate both principles: they are inherently difficult for users to fully understand and they largely limit the control given to the end-user. Thus, there is an opportunity to explore how current design principles apply to the human-computer interaction in interactive machine learning.

Some researchers have started to suggest new principles for designing end-user interaction with general artificial intelligence 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 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 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.

Developing techniques and standards for appropriately evaluating interactive machine learning systems

Although 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. In addition, as our case studies illustrated, some interaction techniques may be appropriate for certain scenarios of use but not others. Evaluations should therefore be careful not to overgeneralize successes or failures of specific interaction techniques. Rather, the scenarios and contexts of use should be generalized to better understand when to apply certain techniques over others.

Leveraging the masses during interaction with machine learning

Most of the case studies in this article focused on a single end-user interacting with a single 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.

Algorithmic problems in interactive machine learning

Research on user interactions with interactive machine learning raises two important technical challenges. First, the requirement for *rapid* model updates often necessitates trading off accuracy with speed. The resulting models are therefore sub-optimal. Although interactive

machine learning can deal with this problem through more iterations, algorithms that are both fast and accurate would improve the quality of learned models and reduce the number of iterations needed to obtain useful models. Second, as some of the case studies described in this article showed, users may desire to interact with machine learning systems in ways that are not supported by existing machine learning methods. Addressing this challenge requires the development of new frameworks and algorithms that can handle different inputs and outputs that are desirable and natural for end-users.

Increasing collaboration across the fields of human computer interaction and machine learning

The inherent coupling of the human and machine in interactive machine learning underscores the need for collaboration across the fields of human-computer interaction and machine learning. This collaboration will benefit human-computer interaction researchers in solving the algorithmic problems discussed above and provide more powerful tools to end-users. In turn, machine learning researchers would benefit by having new methods evaluated with potential users to address practical issues and by developing new frameworks that support realistic assumptions about users.

Finally, we believe that the diversity of perspectives will benefit both communities. For example, when dealing with noisy problems, 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 that end-users induce. Collaboration between these two communities could leverage the benefits of both solutions.

Conclusion

The case studies presented in this article support three key points. First, *interactive machine learning differs from traditional machine learning.* The interaction cycles in interactive machine learning are typically more rapid, focused, and incremental than in traditional machine learning. This increases the opportunities for users to impact the learner and, in turn, for the learner to impact the users. As a result, the contributions of the system and the user to the final outcome cannot be decoupled, necessitating an increased need to study the system together with its potential users.

Second, *explicitly studying the users of learning systems is critical to advancing this field*. Formative user studies can help identify user needs and desires, and inspire new ways in which users could interact with machine learning systems. User studies that evaluate interactive machine learning systems can reveal false assumptions about potential users and common patterns in their interaction with the system. User studies can also help to identify common barriers 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, and in some situations may not be appropriate or desired by end-users. 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, this article 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 towards more effective interactions. We believe this will lead not only to more capable machine learners, but also more capable end-users.

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.

Amershi, S., Fogarty, J., Kapoor, A. and Tan, D. 2010. Examining Multiple Potential Models in End-User Interactive Concept Learning. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2010 (CHI 2010)*, pp. 1357-1360.

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.

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.

Freund, Y., & Schapire, R. E. (1995). A decision-theoretic generalization of on-line learning and an application to boosting. In *Computational learning theory* (pp. 23-37). Springer Berlin Heidelberg.

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*.

Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement learning: A survey. *arXiv preprint cs/9605103*.

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 Multi-agent 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.

Pu, P., & Chen, L. 2009. User-Involved Preference Elicitation for Product Search and Recommender Systems. *AI Magazine*, 29(4), 93.

Rashid, A. M., Ling, K., Tassone, R. D., Resnick, P., Kraut, R., & Riedl, J. 2006. Motivating participation by displaying the value of contribution. In Proceedings of the SIGCHI conference on Human Factors in computing systems (pp. 955-958). ACM.

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., 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.

Su, X., & Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques. *Advances in artificial intelligence*, *2009*, 4.

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.

Ware, M., Frank, E., Holmes, G., Hall, M., & Witten, I. H. (2001). Interactive machine learning: letting users build classifiers. *International Journal of Human-Computer Studies*, *55*(3), 281-292.

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