Designing for Effective End-User Interactive Machine Learning

Saleema Amershi
Computer Science & Engineering
DUB Group
University of Washington
Seattle, WA 98195 USA
samershi@cs.washington.edu

James Fogarty
Computer Science & Engineering
DUB Group
University of Washington
Seattle, WA 98195 USA
jfogarty@cs.washington.edu

CHI'12, May 5–10, 2012, Austin, Texas, USA.
ACM 978-1-4503-1016-1/12/05.

Abstract
Interactive machine learning offers great potential as a tool for scaling end-user interaction with modern data. This position paper first briefly motivates a need for end-user interactive machine learning. We then present our prior work on three interactive machine learning applications: the CueFlik system for web image search, the CueT system for network alarm triage, and the ReGroup system for friend grouping in social networks. Finally, we discuss our interest in this workshop and new potential directions for this area of research.

Introduction
Modern computing is defined by a data explosion that challenges basic assumptions of current interfaces. Although application developers build automated processing and decision-making into applications, this limits end-users to the support conceived and provided by the developer. If we want to give people greater ability to control and manipulate their own data, we need new strategies for enabling effective end-user interaction with that data. Machine learning provides a powerful approach, but little is known about designing for effective end-user interaction with learning-based systems. This is a critical problem: just as the past twenty years have seen the emergence of search, we expect the next twenty years will see machine learning embedded in every aspect of end-user interaction.

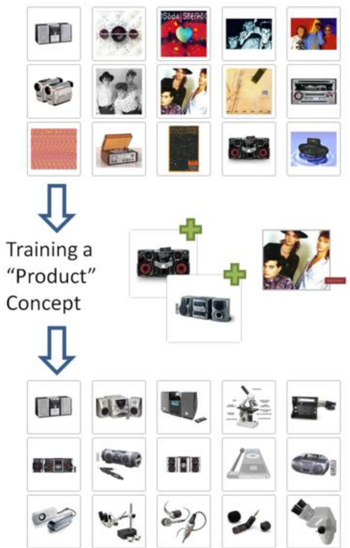

Figure 1. CueFlik is an end-user interactive machine learning system for web image search. End-users create their own content-based filters, which they can apply to the current query and re-use with future queries.

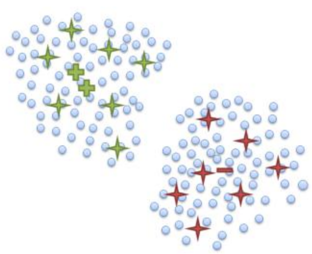

Figure 2. We developed general methods for sampling *overviews* of positive and negative regions. Overviews led end-users to select better examples throughout the interactive machine learning process, resulting in better models.

Background and Prior Research

Our prior research includes design, development, and experimentation in three interactive machine learning applications: the CueFlik system for web image search, the CueT system for network alarm triage, and the ReGroup system for friend grouping in social networks.

CueFlik: Web Image Search

CueFlik examines end-user training of machine learning systems in the context of web image search. Keywords are a fundamentally impoverished method for characterizing images, so modern engines combine keyword queries with content-based filters. Beyond excluding pornography, major engines provide filters for black and white images, clipart, or images that contain faces. But consider that no major engine provides a filter for "product" images (i.e., crisp photos of objects on empty backgrounds). This is not a difficult computer vision problem. Instead, the problem is that no developer can predict and provide all desired filters. CueFlik allows end-users to interactively train a machine learning system to recognize concepts. A person can train a "product" concept to filter images from the query "stereo", then re-use it to filter "phone" or "bike" images. End users asked to try their own ideas created such examples as a "sports action shot" concept from "football" and "basketball" images. We presented CueFlik at CHI 2008 [7].

Iterative feedback is central to interactive machine learning. A person expresses intent, the system provides feedback about what it has learned, and the process iterates until a person is satisfied. In CueFlik, both end-user intent and system feedback are presented through examples. Our CHI 2008 research compared feedback in a single view ranked by positive likelihood with a split view of fewer high-certainty positive and negative examples. The split view led to better models, based on fewer examples, trained in less time [7]. Our analyses suggest the split view naturally steers end-users away from problems of overfitting with noisier examples. Our UIST 2009 research extends this with methods for sampling overviews of positive and negative regions. Prior work has generally focused on examples that are "exemplary" (i.e., most central) or "informative" (i.e., at the boundary), but we showed that our overviews led participants to select better training examples throughout the interactive process, resulting in better models [1]. Despite the overall effectiveness of this feedback, participants sometimes expressed uncertainty about whether to continue providing examples. Our CHI 2010 research therefore re-examined an implicit assumption of prior interactive machine learning. Instead of a hill-climbing process limited to providing *more* training examples, we examined flexible end-user exploration of *different* training examples. We found that end-users naturally adopt undo and history metaphors for examining multiple potential models, are more comfortable exploring a problem, and train better models [2]. These results highlight the importance and difficulty of designing interactive machine learning [3]. Our methods should also generalize beyond CueFlik, as split views, our overview methods, and our approach to undo and history metaphors for examining multiple potential models can be applied in any example-based interactive machine learning system.

CueT: Network Alarm Triage

CueT examines interactive machine learning in the context of network alarm triage. Modern large networks contain thousands of diverse devices, each generating a

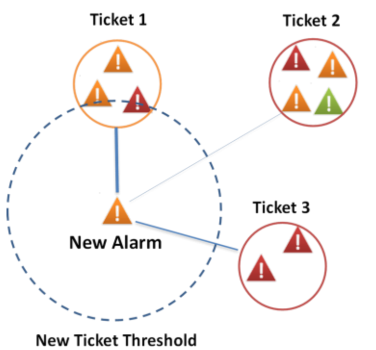

Figure 3. CueT helps operators interactively cluster a stream of network alarms into tickets corresponding to shared underlying causes. In this example, there are three existing tickets. The dashed circle indicates the distance thresholds at which CueT suggests an alarm might be a new ticket. The distance metric and the new ticket thresholds are both learned from operator actions.

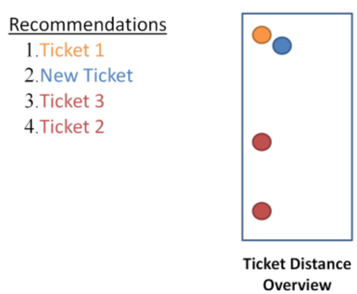

Figure 4. Triage errors can be very problematic, and a naïve ranked list can obscure ambiguity. CueT instead uses a compact confidence visualization to help draw operator attention to ambiguous alarms.

different set of alarms. To manage the errors inherent to automated systems, networks invariably employ human operators for alarm triage. These operators sift through thousands of alarms per day, grouping them according to shared underlying causes. CueT aims to support these operators by continuously learning from their actions and in turn using its learned models to assist in their work. This continuous learning from human feedback is critical to serving the unique needs of a given network and also evolving with that network. We presented CueT at CHI 2011 [5], and we were also invited to present it at IJCAI 2011 [6].

CueT is implemented as interactive clustering of a stream of incoming alarms, where each cluster is called a ticket. When a new alarm arrives, CueT uses a learned distance metric to compare the alarm to existing tickets. It then suggests a set of tickets to which the operator might add the alarm. If the alarm is sufficiently distant from existing tickets, CueT can also suggest creation of a new ticket. Because triage errors can be very problematic and a naïve ranked list can obscure ambiguity, CueT uses a compact visualization of confidence to help ensure operators pay more careful attention to ambiguous alarms. Both the distance metric and the new ticket threshold are learned from operator actions. In a study of operators triaging actual network events, CueT made operator triage 37% faster and also improved the accuracy of operator decisions.

ReGroup: Social Network Friend Groups

ReGroup examines end-user interactive machine learning for creating friend groups on social networks. Existing systems focus on pre-creating groups that are later selected when sharing content. But research in usable security has shown that such groups do not correspond to those described at the time of an actual sharing decision (e.g., a notion of "co-workers" or "close friends" may change depending upon what content will be shared). ReGroup supports creation of groups in the context of sharing. As a group is being created, ReGroup suggests additional members. It also suggests feature-based filters for narrowing a list of friends. ReGroup will be presented at CHI 2012 [4].

ReGroup demonstrates several approaches to effective end-user interaction with machine learning. Feedback about the features in a learned model is presented in the context of familiar faceted browsing techniques.

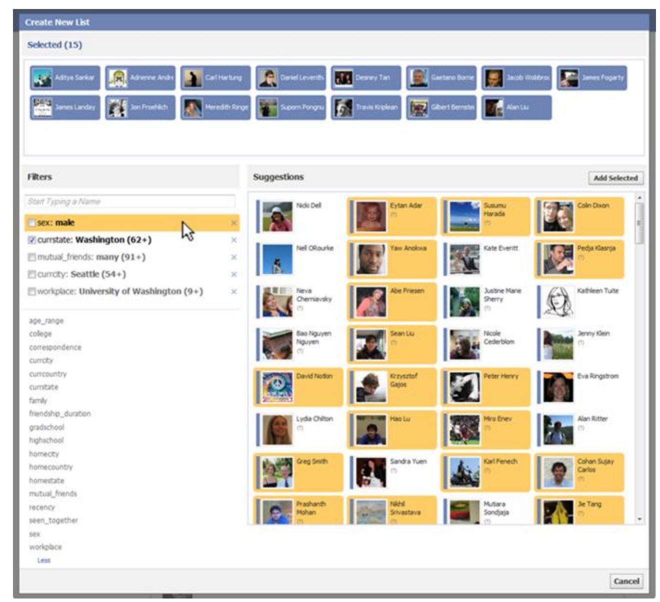

Figure 5. ReGroup uses end-user interactive machine learning to help people create custom, on-demand groups. As a person selects group members, ReGroup suggests additional members and suggests group characteristics as filters for narrowing a friend list (see five suggested filters at the top of Filters area).

Because people generally do not provide ReGroup with negative examples (i.e., explicit indications that a person should not be in a group), ReGroup infers implicit negative examples by observing when people skip over some friend to add others to the group. Participants found ReGroup's methods a powerful complement to existing designs that use alphabetical lists together with keyword search. Specifically, participants found traditional interfaces effective for small groups of known people that can easily be recalled (e.g., immediate family, co-authors of a paper, members of a cross-country team). But ReGroup was preferable and more effective for larger and more homogeneous groups, as it allowed participants to rely upon recognition instead of requiring recall (e.g., childhood friends, friends in particular regions of the world, friends with a professional relationship).

Participation and Future Directions

We look forward to the CHI 2012 Workshop on End-User Interactions with Intelligent and Autonomous Systems.

One of our interests is in discussion of explicit end-user re-use of machine learning systems trained in previous interactions. The success of tagging has shown that meaningful annotations of data can have many uses, but tags are limited by their lack of generalization. We envision future systems where end-users train composable concepts as a part of configuring more sophisticated data processing and manipulation. We also envision sharing such concepts, allowing people to leverage and build upon each other. The ability to interactively define arbitrary distinctions in data is much more powerful than has been explored in existing applications, and we believe there are many important and interesting research questions in this area.

Acknowledgements

We thank all of the additional co-authors on the reviewed work. This work was supported in part by the National Science Foundation under awards IIS-812590, IIS-1016713, and OAI-1028195, by Office of Naval Research award N00014-06-1-0147, by the WRF / TJ Cable Professorship, by Microsoft, and by a Google Faculty Research Award.

References

[1] Amershi, S., Fogarty, J., Kapoor, A., and Tan, D. Overview-Based Example Selection in End-User Interactive Concept Learning. *UIST 2009*, pp. 247-256.

[2] Amershi, S., Fogarty, J., Kapoor, A., and Tan, D. Examining Multiple Potential Models in End-User Interactive Concept Learning. *CHI 2010*, pp. 1357-1360.

[3] Amershi, S., Fogarty, J., Kapoor, A., and Tan, D.S. Effective End-User Interaction with Machine Learning. *AAAI 2011*, Nectar Track, pp. 1529-1532.

[4] Amershi, S., Fogarty, J., and Weld, D.S. ReGroup: Interactive Machine Learning for On-Demand Group Creation in Social Networks. *CHI 2012*, to Appear.

[5] Amershi, S., Lee, B., Kapoor, A., Mahajan, R., and Christian, B. CueT: Human-Guided Fast and Accurate Network Alarm Triage. *CHI 2011*, pp. 157-166.

[6] Amershi, S., Lee, B., Kapoor, A., Mahajan, R., and Christian, B. CueT: Human-Guided Fast and Accurate Network Alarm Triage. *IJCAI 2011*, Invited Paper, pp. 2564-2569.

[7] Fogarty, J., Tan, D., Kapoor, A., and Winder, S. CueFlik: Interactive Concept Learning in Image Search. *CHI 2008*, pp. 29-38.