Interactive Machine Learning Heuristics

Eric Corbett[1,2*]
[2]Digital Media Program,
Georgia Institute of Technology

Nathaniel Saul[1,3†]
[3]Department of Mathematics,
Washington State University

Meg Pirrung[1‡]
[1]Visual Analytics Group,
Pacific Northwest National
Laboratory

Abstract

End-user interaction with machine learning based systems will result in new usability challenges for the field of human computer interaction. Machine learning algorithms are often complicated to the point of being literal black boxes, presenting a unique challenge in the context of interaction with and understanding by end-users. In order to address these challenges, the most relied upon usability inspection method, the heuristic evaluation, must be adapted for the unique end-user experiences that interactive machine learning presents. To address this gap, this paper introduces ten heuristics for interactive machine learning. These heuristics have been developed by distilling design principles from interactive machine learning literature.

Index Terms: Human-centered computing—Visualization—Visualization design and evaluation methods

1 Introduction

Machine learning has enabled many new forms of user experiences and interactions across the landscape of human computer interaction such as self-driving cars, voice assistants and personalized recommendation systems. These advances push machine learning from being an underlying technical infrastructure unaddressed by HCI research to the forefront of user experience and interface design [13, 31, 33]. As such, machine learning is now a new frontier for human computer interaction: as a source of innovation for user experience and design [26, 30] which in turn requires new design methods and research practices [6, 8]. Much of the research into this space falls under purview of *interactive machine learning* (IML).

According to Dudley and Kristensson IML is a new paradigm that seeks to *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 [9].* IML draws from similar logic as 2^{nd} wave HCI [4] which emphasizes the importance of understanding both the agency and context of end users [32]. From this perspective, what is important to the user is often an emergent property of interaction. In contrast to how machine learning has traditionally been undertaken, often overlooking the importance of both the agency and context of end-users [2], IML seeks to leverage end-users in design of machine learning models by enabling the end-user to guide systems to their own needs and purposes.

A key element of IML is the design of interfaces that enable and support co-adaptivity such that the end-users interactions and target model directly influence each others behavior. These interactions between the system and user are emergent properties of the interface, which raises interface design and usability as a key challenges for IML. Acknowledging the challenge these new interactions present, noted machine learning scholar Amershi poses the question:

> How we should design future human interaction with interactive machine learning, much like we have for designing traditional interfaces [2]?

An important step towards answering this challenge will be addressing the gulf between traditional interface evaluation methods [18] and interface evaluation methods suitable for IML [5].

While there is growing acknowledgement of the challenges user experience (UX) and visualization practitioners face while working with machine learning [8, 12, 33, 34]; the method of heuristic evaluation has not been updated to address the challenges of IML. The heuristic evaluation is the most basic tool for improving the usability of an interface [24]. Heuristic evaluation is vital in conditions of limited time and access that many UX practitioners and visualization experts can find themselves in while working with fast-paced and often adversarial development environments. In the constellation of UX methods, heuristic evaluation is often referred to as a *discount usability method* because it can used by practitioners without advanced training, requires no special equipment and requires little time [23]. Of course, heuristic evaluation should be one of many techniques applied in the grand scheme of human centered design but in many cases the only user research done (or rather allowed to be done by the constraints of industry) is the heuristic evaluation. For these reasons, developing a rigorous, empirically tested and theoretically sound heuristic evaluation method suitable for the unique challenges of IML is urgent work for the HCI, visualization and machine learning communities.

2 Heuristics for IML

A heuristic evaluation is an informal expert evaluation technique that relies on a series of *heuristics* (broad, general rules) to aid decision making [23]. The goal of the evaluation is to determine if the interface conforms to the heuristics. By checking the system against a set of heuristics, baseline usability can be achieved earlier on in the system design process prior to more formal, time consuming evaluations with end-users.

The set of heuristics developed by Nielsen and Molich [24] are the most well-known and used throughout HCI. These original ten heuristics were developed in the early 90s based on the most common usability issues known across various desktop computing applications. However, the range and form of both hardware and software has grown rapidly. As such, there have been many new sets of heuristics developed [19]. These new heuristics are typically some modification of the original ten such that they are amenable to the intricacies of a new domain (e.g. ambient displays [20], interactive televisions [7], ubiquitous computing [28], visual analytics [27]). Similarly, our work addresses the various limitations of the original ten heuristics when applied to unique aspects of IML.

Researchers have noted how IML challenges many of the assumptions of usability that the original heuristics were meant to address [2, 3, 16]. To illustrate this point, consider how the interaction principle that underlies the systems Nielson's heuristics were intended to address is *direct-manipulation*: that objects of interest for the user should be visible and interactable in a simple, direct manner analogous to how the user might in real life [17]. Nielson's heuristics 1, 2, 3 and 4 attempt to enforce direct manipulation. In contrast, some

[*]e-mail: ecorbett@gatech.edu
[†]e-mail: nathaniel.saul@wsu.edu
[‡]e-mail: meg.pirrung@pnnl.gov

of the keys aspects of IML are not available to direct manipulation given that some types of machine learning models, specifically deep learning models, cannot be directly manipulated [21]. Moreover, an IML system will evolve as it receives additional user feedback which may make its behaviors difficult to predict which violates heuristics 1 and 3. Furthermore, this evolution may occur in such a way that is not intuitive or clear which would violate heuristics 1, 3, 4, and possibly 5.

To address some of the breakdowns noted above, we have developed ten heuristics specific to IML systems. Our method for developing these heuristics follow the guidelines for generating new domain specific heuristics described by [14]. We began by first identifying the key elements of IML defined within two large review papers of multiple IML systems [2, 9]. Next, we extracted the most salient design guidelines, usability challenges, and interaction techniques of IML user interaction within [2, 9, 16, 31]. We then grouped this extracted information into categories based on similarity while omitting redundancies and translating more abstract, higher level design language into tangible action statements suitable for heuristic evaluation procedures. We compared and contrasted these action statements with Nielsen's general set of heuristics drawing from our collective experiences designing and evaluating IML systems to modify language and scope. Based on these comparisons and conversations we iterated and refined the action statements into ten heuristics.

The specific interpretation and relative importance of these heuristics are likely to vary depending on the level of involvement expected of the user and on the complexity of the required functionality. Compare, for example, a user seeking to instruct a content suggestion service to obtain better recommendations versus a user seeking to train a robot to perform some function in response to a given input. This variability across applications frustrates efforts to obtain concise heuristics generalizable to all IML applications. With that in mind, these heuristics attempt to cover a wide range of IML functionality. We categorized the heuristics under three categories: *Model Input*, *User Feedback*, and *Interaction Design*.

2.1 Model Input

The most fundamental component of an IML system is to enable user interaction with the model. To accomplish this, users must provide some input into the model. Heuristic 1 summarizes the need for users to provide general guidance for the goals and intention of the model, while Heuristic 2 encourages systems to adopt nuanced feedback mechanisms. To further improve model input, Heuristic 3 encourages the interface to consider intent rather than input in order to encourage user engagement and understanding.

Heuristic 1: Enable the User to Steer the Model

Explanation: The interface should enable the user to iteratively steer the model towards a desired concept through the interaction techniques available and the visual feedback presented.

Example: Model steering for a facial expression recognizer towards the concept *angry face* could be achieved by interface features that allow the user to iteratively train the classifier. Guidance to the user in this steering task could be achieved with the following the instructions: When you show examples of *angry face*, vary them as much as possible. Appropriate framing of the task and simple guidance can positively influence user performance and model quality.

Benefit: By creating a tight coupling between the system and the user, model steering engenders a sense of user control.

Problem: In some cases, numerous user actions may be required before the model reflects a desired change. An application that cannot respond in a timely manner to user steering is a potential cause of significant frustration.

Heuristic 2: Enable the User to Provide Feedback that Improves Concept Quality

Explanation: Allow the user to provide feedback on specific instances by (re)assigning labels, selecting or re-weighting features, generating new samples, or adjusting costs matrices.

Example: An image processing tool could allow the user to draw directly on images to guide the training of an image classifier. The user reviews the current classifier performance based on its classification of image regions and provides more feedback by drawing on the image if necessary to improve concept quality.

Benefit: Enabling the user to provide feedback at the concept level will improve the overall quality of the model while also furthering the sense of control over the system.

Problem: Generally, the preference for the user is for more complete and specific feedback methods. It is important to balance the user's desire for control over behavior and providing exhaustive machine-centric knobs. Finding the right balance between adequate user control and confusing algorithm level parameter adjustment is a difficult problem.

Heuristic 3: Capture Intent Rather than Input

Explanation: What the user does is not always the same as what the user intends. Therefore, the interface should help to extract user intent from potentially noisy input actions where possible (and appropriate).

Example: An interactive machine learning application developed for building custom social networks could capture the user's intent after they skip past contacts by explicitly indicating that these contacts should be labelled as negative samples.

Benefit: Creative ways of capturing intent can provide rich interaction experiences to drive model behavior.

Problem: Users and their tasks are often dynamic and unpredictable. Capturing current intent will often be a moving target.

2.2 User Feedback

User feedback can come in a variety of levels. Heuristic 4 encourages systems to provide general or global model feedback. The proper method is model and system dependent, but often manifest in the form of model scores or confusion matrices. More granular feedback, as discussed in Heuristic 5, is a topic of research for visualization experts. Modern techniques such as LIME [25] or saliency maps [29] should be incorporated into IML systems so users obtain a more nuanced understanding of model behavior. Both of these feedback mechanisms should address Heuristic 6 by providing feedback in as natural way as possible. Considerations for how to visualize feedback during the interactions is of particular interest to visualization researchers [11].

Heuristic 4: Support User Assessment of Model Quality

Explanation: Users need to be able to assess the quality of the current state of the model. Quality can be in terms of coverage, prediction accuracy, or confidence.

Example: A social network group creation tool could present filters generated based on features in the model. User interaction with these filters would provide insight on the patterns that were being exploited by the model and thus serve the dual purpose of allowing the user to assess the model as well as their intended function as an interaction element.

Benefit: Supporting assessment of model quality is vital to scaffold users ability to develop strategies that iteratively work towards desired levels of model quality.

Problem: Depending on the context, *quality* can be a nebulous concept. Determining the proper metrics for the application can be difficult and can change depending on the user.

Heuristic 5: Provide Instance-Based Explanations to the User

Explanation: Provide human-readable illustrations of the learned concept. This can allow users to understand the model predictions in a specific instance of model failure or success.

Example: The classification of a particular message as being related to the topic hockey might be accompanied by examples of other passages about hockey or the explanation this message has many more words related to Hockey than other sports.

Benefit: Understanding why the model fails in a particular instance may help the user determine the most appropriate feedback strategy.

Problem: The most appropriate means of illustration will vary based both on data, application type and on the level of understanding possessed by the user.

Heuristic 6: Support Rich, Natural Feedback

Explanation: People want to provide feedback naturally, rather than be forced to interact in limited, system-centric ways. Support rich, user-centric feedback.

Example: A text classification system for email messages could support a rich variety of user feedback mechanisms to improve the classifiers performance: suggesting alternative features to use, adjusting the importance or weight given to different features, or modifying the information extracted from the text.

Benefit: Providing rich natural feedback mechanisms will improve user engagement and motivation with model steering tasks.

Problem: Care must be taken not to overwhelm the user with too many feedback options. Novel interactions are potentially rich, but equally unfamiliar.

2.3 Interaction Design

The following heuristics provide guidelines for connecting the two-way user-model communication channels into an interactive loop. Heuristic 7 suggests that all model inputs and responses should be explicit. Heuristic 8 encourages systems to maintain mechanisms for rewinding models when users make mistakes. This process facilitates Heuristic 9 preventing significant errors and together they encourage a sense of confidence that the user is free to explore the model and data space. The final heuristic, Heuristic 10, stipulates the importance of documentation when users are both learning a system and exploiting it to its complete potential.

Heuristic 7: Make Interactions and Constraints Explicit

Explanation: Any user interaction that influences model behavior (and/or the constraints in doing so) should be made explicit.

Example: Many recommender systems exploit user input for dual purposes, i.e., a *like* provides a signal of interest to a social group but may also inform future recommendations. While dual-purpose input with unclear objectives may be suitable for recommender systems, generally, being explicit when requesting user input, whether singular, or multi-purpose, is preferable.

Benefit: The degree to which a user understands their ability and limitations in addressing model behavior may have a significant impact on their satisfaction with the interface and the refinement actions with which they proceed.

Problem: The degree to which the user needs to understand how their interactions are impacting the system will need to be balanced against the amount of information required to do so as too much information will can be confusing and detrimental to performance.

Heuristic 8: Promote Trial-and-Error based Model Exploration

Explanation: The process of steering and refining a model is better served if a sense of exploration is promoted within the user. Provide revision mechanisms and history information to support the user to actively explore the model space.

Example: A visualization of model improvement or degradation along with *undo* functionality could guide user efforts to refine the model.

Benefit: Promoting trial-and-error based exploration leverages human insight and promotes engagement.

Problem: Directing too much user attention towards model exploration may not always correlate well with the users overall goal. Additionally, providing robust mechanisms for *undo* functionality could be laborious depending on the underlying machine learning techniques used.

Heuristic 9: Error Prevention

Explanation: Users are often imprecise and inconsistent. They may not stick to a cohesive concept, and additionally introduce errors and bias. All of these common user flaws will have a negative effect on trained model quality. Careful design of the interface, both in terms of the information presented and guidance, can help prevent these user errors.

Example: In order to prevent user errors in labeling tasks, targeted information can be provided when a label is requested. The information provided to the user could be 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, or its uncertainty in this prediction.

Benefit: Errors can have a lasting impact on the quality of the system and interactions. Minimizing errors will improve both the process and the final outcome. Additionally, if the user understands that they are in part responsible for errors, then they may be more forgiving in their perception of the system.

Problem: High complexity of interaction with IML makes it difficult for users to perceive or prevent their own errors.

Heuristic 10: Help and Documentation

Explanation: Sometimes users will want to learn how to provide nuanced feedback to steer the system. Therefore, tutorials should describe how various controls and actions will impact the learner.

Example: An interactive tutorial that walks users through interface features and strategies to steer the model or assess model quality.

Benefit: Users trust of a system is based on how well they understand it. Many users will be unfamiliar with the IML interaction paradigm so help and documentation will be vital.

Problem: Balance will be needed between explaining the core concepts of machine learning that underlie interactions within an IML system and not overwhelming the user with technical details.

3 Discussion

Taken together, these ten heuristics cover the essential elements of IML user interface design as outlined in [2, 9]. There are many ways to utilize these heuristics. To start, the heuristics can be used to guide early system development choices such as selection of machine learning algorithm, explanatory visualization technique, or interface design paradigm. For example, some standard machine learning algorithms have relatively easy to visualize underlying architecture whereas others such as deep learning algorithms do not. In this case, using the heuristics informs how an early decision to use a deep learning algorithm will later require more nuanced methods of model explanation at the interface level.

Using these heuristics in an evaluation of an existing system should allow designers to achieve baseline usability for user experience with IML functionality. This is the primary function of heuristic evaluation: to identify and address usability challenges prior to user interaction. This allows for later evaluations with users to focus on more complex issues of user experiencesuch as trust [10], decision-making [15] and overall satisfaction [22]. These more complex aspects of user experience can be difficult to assess if a myriad of usability issues exist.

Finally, these heuristics can be referenced later on during formal evaluations with end-users. For example, in our own ongoing work, we have used the ten heuristics to categorize the types of usability issues that have emerged in our user studies. Used in this manner, the heuristics provide both a lens and language for understanding users interactions during studies.

Our current work aims to formally evaluate these heuristics in a comparative study against Nielson's original set. We are interested in quantity and type of errors found, as well as the experience of evaluators in successfully applying the heuristics. We will use the results to provide empirical insight on the conversation we have started in this work, as well as improve our proposed set of interactive machine learning heuristics.

4 Conclusion

As Dudley and Kristensson noted:

> Machine learning techniques are slowly creeping into the lives of non-expert users. Enabling users to efficiently interact with such algorithms is likely to be a key design challenge in the coming decade [9].

A key step towards addressing this challenge will require researchers across the visualization, machine learning, and HCI communities to develop evaluation practices suitable for the unique usability issues presented by IML. As an approach, IML is vital to address the growing ubiquity of systems with underlying machine learning techniques that people are using without understanding, often leading to adverse results. Using these heuristics will increase user understanding and engagement with machine learning systems. The clarity, control, and power afforded to users in IML might help to address the ongoing challenges of bias, transparency, and accountability being faced in the larger machine learning community [1].

References

[1] http://www.fatml.org/.

[2] S. Amershi, M. Cakmak, W. B. Knox, and T. Kulesza. Power to the People: The Role of Humans in Interactive Machine Learning. *AI Magazine*, 2014. doi: 10.1609/aimag.v35i4.2513

[3] A. F. Blackwell. Interacting with an Inferred World: The Challenge of Machine Learning for Humane Computer Interaction. *Aarhus Series on Human Centered Computing*, 2015. doi: 10.7146/aahcc.v1i1.21197

[4] S. Bødker and Susanne. When second wave HCI meets third wave challenges. In *Proceedings of the 4th Nordic conference on Human-computer interaction changing roles - NordiCHI '06*, pp. 1–8. ACM Press, New York, New York, USA, 2006. doi: 10.1145/1182475.1182476

[5] N. Boukhelifa, A. Bezerianos, and E. Lutton. Evaluation of Interactive Machine Learning Systems. *ArXiv*, pp. 1–20, 2018.

[6] J. Burrell. How the machine thinks': Understanding opacity in machine learning algorithms. *Big Data & Society*, 2016. doi: 10.1177/2053951715622512

[7] C. A. Collazos, C. Rusu, J. L. Arciniegas, and S. Roncagliolo. Designing and evaluating interactive television from a usability perspective. In *Advances in Computer-Human Interactions, 2009. ACHI'09. Second International Conferences on*, pp. 381–385. IEEE, 2009.

[8] G. Dove, K. Halskov, J. Forlizzi, and J. Zimmerman. UX Design Innovation: Challenges for Working with Machine Learning as a Design Material. In *CHI '17 Proceedings of the 2017 annual conference on Human factors in computing systems*, pp. 278–288, 2017. doi: 10.1145/3025453.3025739

[9] J. J. Dudley and P. Ola Kristensson. A Review of User Interface Design for Interactive Machine Learning. *ACM Trans. Interact. Intell. Syst*, 8(8), 2018. doi: 10.1145/3185517

[10] M. T. Dzindolet, S. A. Peterson, R. A. Pomranky, L. G. Pierce, and H. P. Beck. The role of trust in automation reliance. *International Journal of Human Computer Studies*, 2003. doi: 10.1016/S1071-5819(03)00038-7

[11] M. El-Assad, D. H. Chau, A. Perer, H. Strobelt, and F. Vigas. VISxAI Workshop at IEEE VIS 2018.

[12] K. Fisher. Predictably Smart: Machine learning works best when users don't have to think twice.

[13] M. Gillies, B. Lee, N. D'Alessandro, J. Tilmanne, T. Kulesza, B. Caramiaux, R. Fiebrink, A. Tanaka, J. Garcia, F. Bevilacqua, A. Heloir, F. Nunnari, W. Mackay, and S. Amershi. Human-Centred Machine Learning. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '16*, 2016. doi: 10.1145/2851581.2856492

[14] S. Hermawati and G. Lawson. Establishing usability heuristics for heuristics evaluation in a specific domain: Is there a consensus? *Applied ergonomics*, 56:34–51, 2016.

[15] T. Hirsch, C. Soma, K. Merced, P. Kuo, A. Dembe, D. D. Caperton, D. C. Atkins, and Z. E. Imel. "it's hard to argue with a computer": Investigating psychotherapists' attitudes towards automated evaluation. In *Proceedings of the 2018 Designing Interactive Systems Conference*, DIS '18, pp. 559–571. ACM, New York, NY, USA, 2018. doi: 10.1145/3196709.3196776

[16] E. Horvitz. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems the CHI is the limit - CHI '99*, 1999. doi: 10.1145/302979.303030

[17] E. L. Hutchins, J. D. Hollan, and D. A. Norman. Direct manipulation interfaces. *Human-computer interaction*, 1(4):311–338, 1985.

[18] R. Jeffries and H. Desurvire. Usability testing vs. heuristic evaluation. *ACM SIGCHI Bulletin*, 24(4):39–41, oct 1992. doi: 10.1145/142167.142179

[19] C. Jimenez, P. Lozada, and P. Rosas. Specific-Domain Usability Heuristics: Are they really necessary? *Revista Romana de Interactiune Om-Calculator*, 10(1):1–24, 2017.

[20] J. Mankoff, A. K. Dey, G. Hsieh, J. Kientz, S. Lederer, and M. Ames. Heuristic evaluation of ambient displays. In *Proceedings of the conference on Human factors in computing systems - CHI '03*, p. 169. ACM Press, New York, New York, USA, 2003. doi: 10.1145/642611.642642

[21] G. Marcus. Deep learning: A critical appraisal. *arXiv preprint arXiv:1801.00631*, 2018.

[22] J. McCarthy and P. Wright. *Technology as experience*, vol. 11. The MIT Press, 2004. doi: 10.1145/1015530.1015549

[23] J. Nielsen and Jakob. Finding usability problems through heuristic evaluation. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '92*, pp. 373–380. ACM Press, New York, New York, USA, 1992. doi: 10.1145/142750.142834

[24] J. Nielsen and R. Molich. Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems Empowering people - CHI '90*, pp. 249–256. ACM Press, New York, New York, USA, 1990. doi: 10.1145/97243.97281

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

[26] A. Sarkar. Constructivist Design for Interactive Machine Learning. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '16*, 2016. doi: 10.1145/2851581.2892547

[27] J. Scholtz. Beyond usability: Evaluation aspects of visual analytic environments. In *Visual Analytics Science and Technology, 2006 IEEE Symposium On*, pp. 145–150. IEEE, 2006.

[28] J. Scholtz and S. Consolvo. Toward a framework for evaluating ubiqui-

tous computing applications. *IEEE Pervasive Computing*, 3(2):82–88, apr 2004. doi: 10.1109/MPRV.2004.1316826

[29] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[30] S. Stumpf, V. Rajaram, L. Li, M. Burnett, T. Dietterich, E. Sullivan, R. Drummond, and J. Herlocker. Toward harnessing user feedback for machine learning. In *Proceedings of the 12th international conference on Intelligent user interfaces - IUI '07*, 2007. doi: 10.1145/1216295.1216316

[31] S. Stumpf, V. Rajaram, L. Li, W. K. Wong, M. Burnett, T. Dietterich, E. Sullivan, and J. Herlocker. Interacting meaningfully with machine learning systems: Three experiments. *International Journal of Human Computer Studies*, 2009. doi: 10.1016/j.ijhcs.2009.03.004

[32] L. A. Suchman. *Plans and situated actions: The problem of human-machine communication*. Cambridge university press, 1987.

[33] Q. Yang, N. Banovic, and J. Zimmerman. Mapping Machine Learning Advances from HCI Research to Reveal Starting Places for Design Innovation. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, pp. 1–11, 2018. doi: 10.1145/3173574.3173704

[34] Q. Yang, A. Scuito, J. Zimmerman, J. Forlizzi, and A. Steinfeld. Investigating How Experienced UX Designers Effectively Work with Machine Learning. In *Proceedings of the 2018 on Designing Interactive Systems Conference 2018 - DIS '18*, pp. 585–596. ACM Press, New York, New York, USA, 2018. doi: 10.1145/3196709.3196730