Interactive Machine Learning for End-User Innovation

Francisco Bernardo, Michael Zbyszyński, Rebecca Fiebrink and Mick Grierson

Goldsmiths University of London
Department of Computing
London, UK

Abstract

User interaction with intelligent systems need not be limited to interaction where pre-trained software has intelligence "baked in". End-user training, including interactive machine learning (IML) approaches, can enable users to create and customise systems themselves. We propose that the user experience of these end users is worth considering. Furthermore, the user experience of system developers–people who may train and configure both learning algorithms and user interfaces to be used by end users–also deserves attention. We additionally propose that IML can improve user experiences by supporting user-centred design processes; at the same time, there is a further role for user-centred design to play in improving interactive and classical machine learning systems. We have developed this approach in the context of the European Commission-funded project, RAPID-MIX, where they are embodied in the design of a new User Innovation Toolkit.

Introduction

When considering the user experience of machine learning systems, it is important to consider the experiences of users designing, training, evaluating, and refining machine learning systems. Sometimes, such users are "end users" of software tools–artists, musicians, people with disabilities, hackers and makers, people applying machine learning to "quantified self" monitoring or to creating new smart home applications. Interactive machine learning (IML) approaches can enable such users to create and customise machine learning applications. Fails and Olsen 2003 define IML as a new machine-learning paradigm with a workflow that features rapid cycles of training and corrective feedback of classifiers with human supervision. In the simplest case, users can interact with the machine learning process by providing training examples (e.g., examples of a human action or activity, alongside the label that a classifier should apply to that action/activity).

In this paper, we describe how IML can support new types of end-user customisation, which can be understood in terms of end-user programming (Lieberman et al. 2006) or as user innovation toolkits (Von Hippel 2005). We describe considerations and challenges for supporting user experience in

this context. Further, we describe a similar context, that of system developers employing machine learning to create intelligent systems for use by others. These developers are also "users" of machine learning tools, and the tools they use impact the systems that can be built and the types of experiences that are possible for their end users.

We also describe how interactive machine learning and user-centred design processes may inform each other. User-Centred design (UCD), which was originally coined by Norman and Draper 1986, is a process based upon the understanding of users, their tasks and environments; it involves the user at an early stage of project development and throughout the project lifetime. In UCD, the design is driven and refined throughout an iterative cycle of development and user-centred evaluation. Interactive machine learning provides a set of mechanisms to support user-centred design, by making it possible to translate users' demonstrations or observations of user actions into training examples that are used to build or refine a new technology. At the same time, user-centred design can and should play a role in the design of interactive and classical machine learning systems, whether aimed at developers or end users.

The final section of our paper describes RAPID-MIX, a design project that aims to place next-generation machine learning tools in the hands of end users in the form of a User Innovation Toolkit. We describe the aims of this project and the methodologies we have employed to understand and improve the user experiences of both end users and professional developers using machine learning to build new interactive systems.

Claims

Interactive machine learning can enable end users to customise systems.

As described above, IML can make it possible for end users to create customised systems. Typically, IML users provide training examples or labels iteratively, or adjust learning parameters to "steer" a machine-learned model toward a desired behaviour. A user interface for IML may enable users who have domain expertise (but possibly no machine learning expertise) to provide training examples (Fiebrink 2011; Fails and Olsen Jr 2003), adjust misclassification costs (Kapoor et al. 2010), adjust weightings of component clas-

sifiers in an ensemble system (Talbot et al. 2009), or take other actions. This approach can be used to create simple computer vision classifiers (Fails and Olsen Jr 2003), new sensor-based interactions (Hartmann et al. 2007), customised gestural controllers including new musical instruments (Fiebrink 2011), customised alerts (Amershi et al. 2011), and potentially many other bespoke systems.

Amershi, Cakmak, Knox, and Kulesza 2014 describe the IML workflow as more rapid, more focused and incremental when compared to classic machine learning (CML), which can be a laborious, and sometimes inefficient process. IML can also offer users without machine learning or programming expertise an effective way to customise systems, since their natural interactions can be used as training examples, and organised through a user interface. The iterative nature of IML means that these same actions can be used to improve systems. For instance, a user can provide additional examples of an action that was misclassified by a trained model, along with the correct label for this action, and reasonably hope that the next model trained on the augmented training set may improve its performance on this type of action.

IML can thus be understood as a tool for end-user programming, as well as a way to "democratise" machine learning, making the benefits of learning algorithms realisable by a wider range of people. Talbot et al. observe that "a critical human-computer interaction challenge is to provide adequate tools to allow non-experts to wield [machine learning] techniques effectively. In order to design these tools, we must start at current best practices in applied machine learning and identify tasks that can be supported or augmented by an effective combination of human intuition and input with machine processing" (p. 1283).

IML can therefore be seen as a set of techniques to enable end users to design new interactions with building blocks provided by industry. For instance, it allows users to create bespoke classifiers or controllers from sensors, and to use these to control systems for gaming, music, smart homes, etc. And such as the seminal example of user innovation toolkits for custom integrated circuit (IC) design industry (Hippel 2001) – which solved a great inefficiency problem, i.e failing to understand the user needs accurately and in detail at the start of a product design project, by allowing users to design their own circuits by themselves. An IML user innovation toolkit can provide for users to customise their designs while addressing inefficiencies in software development, in an analogous way that software design tools were used to do the same for hardware development in the IC industry.

Based on work by Fiebrink and collaborators (Fiebrink, Cook, and Trueman 2011; Fiebrink et al. 2010; Fiebrink 2011), we argue that using IML for user innovation toolkits can provide the following advantages:

- IML can be used by people without programming or machine learning expertise

- IML can facilitate rapid prototyping and iterative refinement activities which are known to be important to design of new systems

- Providing training examples can be a direct and effective way for a user to communicate the desired behaviour of a system, particularly when designing a system to recognise or respond to human actions. Even expert programmers may have difficulty expressing in program code how an embodied activity is to be analysed, but may be able to easily demonstrate an example of the activity.

At the same time, designing effective approaches for end-user innovation with IML presents several challenges, including:

- It may be hard for users to reason how well a system will work in the future. Fiebrink et al 2011 show that conventional metrics of quality computed from the training data (e.g., cross-validation) aren't appropriate for IML systems in which the user employs training data to steer the model behaviour. It may not be clear how to assess whether a model is trustworthy, how to identify likely model mistakes, or how to do these efficiently (i.e., without experimentally feeding the model new data and observing its response).

- It may be difficult for users to select or construct appropriate features. Many real-world applications cannot be built easily from raw data, but rather require some processing of the data in order for learning to be possible. For instance, building new interactions with sensors may require smoothing, segmentation, or analysis of statistics over time windows. Such feature engineering can be difficult even for users with programming and signal processing expertise.

- It may be difficult to collecting appropriate training data and understand this data, especially when the user's goal is to build a system that generalises well to new users or environments whose data he/she cannot easily replicate for training or testing.

- It may be difficult for end users to connect machine learning tools to other tools of interest (e.g., existing systems for home automation, music, activity sensors, etc.) Existing sensors, hardware, and software may not use interoperable or open communication protocols.

- Not all designs a user might imagine will be learnable with the available algorithms, features, and data. Users may struggle to understand what is learnable and what is not.

IML can also be useful for professional developers

Even when the "end user" interacts with a pre-trained intelligent system, the user experience of the developer deserves consideration. Both the set of supported learning algorithms and user interfaces for employing those algorithms impact the developer experience. Just as with end-user development, IML can allow developers to build new systems by demonstration. This can result in a faster development process. When the goal is to build a system that responds to embodied interaction (e.g., human actions sensed with sensors), IML can also result in systems that respond more accurately to the desired interactions.

Improving the interfaces used by developers can make the process of training machine learning-based systems more efficient and effective (e.g., (Patel et al. 2010; Amershi et al. 2015)). The developer/designer often encounters the same challenges identified for end users above. Furthermore, the developer/designer may also have to deal with a wider range of choices of algorithm types. Different types of algorithms afford different end user interactions (e.g., considering sensor analysis: classification, regression, and temporal modelling such as HMM all enable different modes of end user interaction. Variants of these algorithms have also been designed specifically to support interactive training or new types of user interaction (e.g., hierarchical HMMs (Françoise et al. 2014)). Designers/developers may also struggle with feature engineering, even though they may have more signal processing and programming expertise. They may also encounter substantial challenges in understanding and cleaning training datasets collected from target users.

Another significant challenge is that designers of intelligent systems that allow end-user customisation further must choose which interactions to expose to end users and how to expose them in a user interface. Amershi 2012 argues that there are a lack of established principles and proven techniques for designing interaction with machine learning, and that knowledge should advance from ad-hoc solutions into generalized understanding about the IML process.

IML can support user-centred design.

Typically, user-centred design (UCD) activities involve iterative cycles of working with users to identify design directions, implementing prototype designs, and getting feedback on those designs from users. IML offers a way to accelerate this iteration for the design of systems that respond to human gestures or actions, or to other events that can be easily simulated or demonstrated by end users. Training examples can be collected directly from user demonstrations and used to instantiate new designs on the spot, which can then be used to elicit feedback. Or, developers can translate observations of users in UCD activities into training examples for the developed system. We have previously used both approaches to develop new gestural musical instruments for people with disabilities (Katan, Grierson, and Fiebrink 2015).

User-centred design should be applied to IML & CML

User-centred design can and should be applied to understanding and improving usability of interactive and classical machine learning, by both end users and developers. The challenges enumerated in the sections above cannot be met by algorithmic or technical innovation alone. Rather, meeting these challenges requires understanding of users' practices and goals. UCD processes have been used in much of the work described above (e.g., (Fiebrink et al. 2010; Patel et al. 2010; Amershi et al. 2014)) to motivate the utility of machine learning systems and improve their usability and effectiveness.

Amershi et al. 2014 highlight the importance of studying users of machine learning applications based on the tighter link that is established between learning systems and their users. They found that end users' goals had a significant impact on the success of the functionality that was designed for and provided by the system. The considerations about whether end users wanted to train an accurate or reusable model, or whether they were interacting directly with machine learning, for instance, evidenced the need for different design requirements, such as providing a greater or lesser number of alternative mechanisms for system feedback and end user control, different levels of guidance, and of complexity in explanations about the quality of the model.

We argue that these questions are still relevant even when the "interface" employed by the user is a software API, as is the case with many developers who are embedding machine learning into new systems. Some existing research considers the user experience of programming languages, IDEs, and APIs (e.g., (Ko and Riche 2011)), but future work should consider the challenges that are unique to working with machine learning.

RAPID-MIX

We are currently participating in a large research project involving the application of UCD methodologies to two groups of users—end users and developers—to improve tools for applying machine learning and, by extension, the interactive systems built with these tools. *Real-time Adaptive Prototyping for Industrial Design of Multimodal Interactive and eXpressive technologies* (RAPID-MIX) is an Innovation Action funded by the European Commission under the Horizon 2020 program. The RAPID-MIX consortium aims to accelerate the production of the next generation of "MIX" technologies by producing hardware and software tools for rapid prototyping and product development and placing them in the hands of end users in the form of a User Innovation Toolkit (Flowers, von Hippel, De Jong and Sinozic 2010) .

The members of the RAPID-MIX consortium, which include three European research labs and five creative small and medium enterprises (SMEs), have established research portfolios in the design and evaluation of embodied and wearable human-computer interfaces for creative and music technology. We have also developed and accumulated a significant portfolio of technologies for multimodal and expressive interaction, which includes tools for IML, music information retrieval and digital signal processing, as well as cloud-based repositories for storing and visualizing audiovisual and multimodal data.

To facilitate the design of new tools for end users and developers to employ machine learning, we have developed a UCD framework that provides a set of guidelines for engaging in UCD research actions. These *UCD actions* attempt to answer and refine key questions throughout the project lifecycle. As part of our development effort, we have performed multiple UCD actions such as:

- Co-design workshops with academic and SME project partners, 2015

- Hack-a-thons and innovation challenges at Sonar Festival in Barcelona, 2015 & 2016

- Workshops and interviews at Maker Faires in Lisbon and Rome, 2015

These UCD actions contributed to define the design space of the toolkit based on insights that were obtained from:

- assessing and aligning the consortium partners' thinking through ideation, identification of scenarios and design challenges

- probing the users with early technologies

- prototyping the integration of these technologies

- defining an initial set of target users and contexts of use, and evolving gradually to a more scoped definition based on user engagement

- gathering of direct insights from the future end users of the toolkit

Most importantly, we were able to deploy our technology probes and prototypes with users that fitted this composite and gradually scoped profile, and obtain feedback from them. The effort of intertwining several rounds of rapid prototype development with user engagement uncovered design problems and specific challenges for the RAPID-MIX consortium. The results of these actions informed the design of the toolkit, allowing it to fulfil the needs, goals and values of prospective users.

Reports on our UCD methodologies and our design guidelines can be found on our website at: http://rapidmix.goldsmithsdigital.com/downloads/

Conclusion

We have argued that IML can enable people (including both end users and developers) to customize interactive systems and can support user-centred design. Furthermore, UCD can and should be applied to the process of creating machine learning applications of all kinds. These positions are informed by, and inform the RAPID-MIX project, which is ongoing. By engaging with members of the user experience design, service design, HCI, HRI and AI communities at this particular AAAI Symposium, we hope to connect with complementary perspectives and potential users of our toolkit and use that feedback in our next cycle of design and prototyping.

Acknowledgements

References

Amershi, S.; Lee, B.; Kapoor, A.; Mahajan, R.; and Christian, B. 2011. CueT: Human-guided fast and accurate network alarm triage. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 157–166.

Amershi, S.; Cakmak, M.; Knox, W. B.; and Kulesza, T. 2014. Power to the people: The role of humans in interactive machine learning. *AI Magazine* 35(4):105–120.

Amershi, S.; Chickering, M.; Drucker, S. M.; Lee, B.; Simard, P.; and Suh, J. 2015. ModelTracker: Redesigning performance analysis tools for machine learning. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 337–346.

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

Fails, J. A., and Olsen Jr, D. R. 2003. Interactive machine learning. In *Proceedings of the 8th international conference on Intelligent user interfaces*, 39–45.

Fiebrink, R.; Trueman, D.; Britt, C.; Nagai, M.; Kaczmarek, K.; Early, M.; Daniel, M.; Hege, A.; and Cook, P. 2010. Toward understanding human-computer interaction in composing the instrument. In *Proc. of the International Computer Music Conference*.

Fiebrink, R.; Cook, P. R.; and Trueman, D. 2011. Human model evaluation in interactive supervised learning. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 147–156.

Fiebrink, R. 2011. *Real-time human interaction with supervised learning algorithms for music composition and performance*. Ph.D. Dissertation, Princeton University.

Flowers, S.; von Hippel, E.; de Jong, J.; and Sinozic, T. 2010. Measuring user innovation in the uk: The importance of product creation by users.

Françoise, J.; Schnell, N.; Borghesi, R.; and Bevilacqua, F. 2014. Probabilistic models for designing motion and sound relationships. In *Proceedings of the 2014 International Conference on New Interfaces for Musical Expression*, 287–292.

Hartmann, B.; Abdulla, L.; Mittal, M.; and Klemmer, S. R. 2007. Authoring sensor-based interactions by demonstration with direct manipulation and pattern recognition. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 145–154.

Hippel, E. 2001. User toolkits for innovation. *Journal of product innovation management* 18(4):247–257.

Kapoor, A.; Lee, B.; Tan, D.; and Horvitz, E. 2010. Interactive optimization for steering machine classification. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1343–1352.

Katan, S.; Grierson, M.; and Fiebrink, R. 2015. Using interactive machine learning to support interface development through workshops with disabled people. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 251–254.

Ko, A. J., and Riche, Y. 2011. The role of conceptual knowledge in API usability. In *VL/HCC*, 173–176.

Lieberman, H.; Paternò, F.; Klann, M.; and Wulf, V. 2006. End-user development: An emerging paradigm. In *End user development*. Springer. 1–8.

Norman, D. A., and Draper, S. W. 1986. User centered design. *New Perspectives on Human-Computer Interaction*.

Patel, K.; Bancroft, N.; Drucker, S. M.; Fogarty, J.; Ko, A. J.; and Landay, J. 2010. Gestalt: Integrated support for implementation and analysis in machine learning. In *Proceedings*

of the 23nd annual ACM symposium on User interface software and technology, 37–46.

Talbot, J.; Lee, B.; Kapoor, A.; and Tan, D. S. 2009. EnsembleMatrix: interactive visualization to support machine learning with multiple classifiers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1283–1292.

Von Hippel, E. 2005. Democratizing innovation: The evolving phenomenon of user innovation. *Journal für Betriebswirtschaft* 55(1):63–78.