When User Experience Designers Partner with Data Scientists

Fabien Girardin[1], Neal Lathia[2]
[1]BBVA Data & Analytics, Avenida de Burgos, 16D 28036 Madrid, Spain
fabien.girardin@bbvadata.com
[2]Skyscanner, Fore Street, London EC2Y 9DT, United Kingdom
neal.lathia@gmail.com

Abstract

The increasing capacity to capture and feed behavioral data for systems to learn is transforming the design of user experiences. In this paper, we discuss two effects of this emerging toolkit. On the one hand, creating experiences with learning algorithms is pushing designers to consider how users begin, evolve, and end their interactions, which themselves produce and consume data. On the other hand, the design of experiences powered by machine learning is now occurring in new, multidisciplinary teams, which presents a variety of frictions and opportunities for misunderstandings which must be overcome. We discuss the similarities and differences in the methods that designers and data scientists use in their work, and conclude with a series of touch points and principles that partnerships between designers and data scientists can consider for productive relationships.

Introduction

Now, with the emergence of machine learning and service provider's ability to collect granular, long term behavioral data (e.g. interactions or transactions with systems, sensor logs) from users, many new digital services are being designed with the *explicit aim* for them to evolve and adapt as they learn from their users.

Services that learn from their users are characterized by a *feedback loop* (Figure 1); an iterative mechanism that typically offers ways to personalize, optimize, improve or automate services that use an underlying source of data. Behavioral data are fed into the system and algorithms use statistical properties of this data to generate knowledge. An interface then communicates that knowledge to enrich the user's experience. Finally, interactions during this experience create new behavioral data that can be used to retrain the learning algorithm- thus spawning a feedback loop. These services, that create opportunities to design new experiences based on recommendations, predictions or contextualization, are now defining how humans and machines interact.

The design of digital services that are underpinned by a feedback loop is also bringing together a variety of disciplines. In particular, we are witnessing a new practice that requires a tight partnership between designers and data scientists, as systems with feedback loops can only be imagined, built, and improved with a holistic view of the how users' experiences are affected by interactions between data, algorithms, and interfaces.

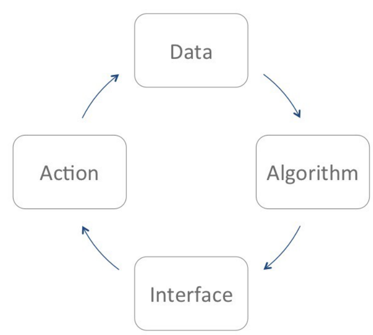

Figure 1. Services that learn from their users are characterized by a feedback loop. Behavioral is fed as context to algorithms that generates knowledge. An interface communicates that knowledge to enrich an experience via an interface. Ideally, that interface seeks explicit user actions or implicit sensor events to create a feedback loop that will feed the algorithm with learning material.

This paper reports on our observations working with user experience designers and data scientists to create experiences with feedback loops. We discuss the different kinds of experiences that can be designed using them, the approaches that designers and data scientists have historically taken when developing systems, and points of friction between the disciplines that need to be addressed in order to build a user experience with a system that learns.

Experiences powered by learning machines

Experiences powered by machines that learn are not linear, nor based on static business or design rules. They evolve according to human behaviors, and are constantly updated as models are fed new streams of data. Each product or service becomes almost like a living, breathing thing; as data scientists at Google would say: “It’s a different kind of engineering” (Levy 2016).

We argue that it is also a different kind of design, driven by the opportunities presented by the novel technologies emerging today. For instance, breakthroughs in machine learning are paving the way to connect humans with machines via the subtle medium of speech. Amazon explains its smart speaker Echo’s braininess as a thing that continually learns and adds more functionality over time. This description highlights how a set of functionalities, and indeed the experience that you can expect to have with this device, will evolve as the machine learns (Figure 2).

There is a very broad, and growing, set of different experiences that can be mediated by machine learning. We have identified several themes in the products coming to life today, and the types of experiences designers can create:

Design for discovery. Recommender systems are now ubiquitous throughout the web, and aim to help users discover the known unknown, or even unknown unknowns, by sourcing knowledge from the crowds of users interacting with the service. This type of experience has a number of challenges that call for design and algorithmic solutions: how can users be encouraged to explore beyond the world that is strictly linked to the user’s profile? How can users control, reshape, and reset the aspects of their profile that influence what they discover? Should discovery be purely algorithm-driven, or augmented with input from editors or other humans?

Design for decision making. Information services also increasingly provide visualizations and interactions aimed to help users make decisions. Algorithms in this domain need to learn to be precise, simply because they often rely on datasets that only give one, possibly incomplete, perspective of reality. Designers can help to overcome poor quality data, by letting users (implicitly or explicitly) reflect on the quality of the data at hand.

Design for uncertainty. Traditionally, the design of computer programs follows a deterministic path, with an explicit set of concrete and predictable states that can be translated into a workflow. Machine learning algorithms, that are designed to look for patterns within that approximate the rules underlying user behaviors (Hebron 2016), will often produce results with varying levels of recall or

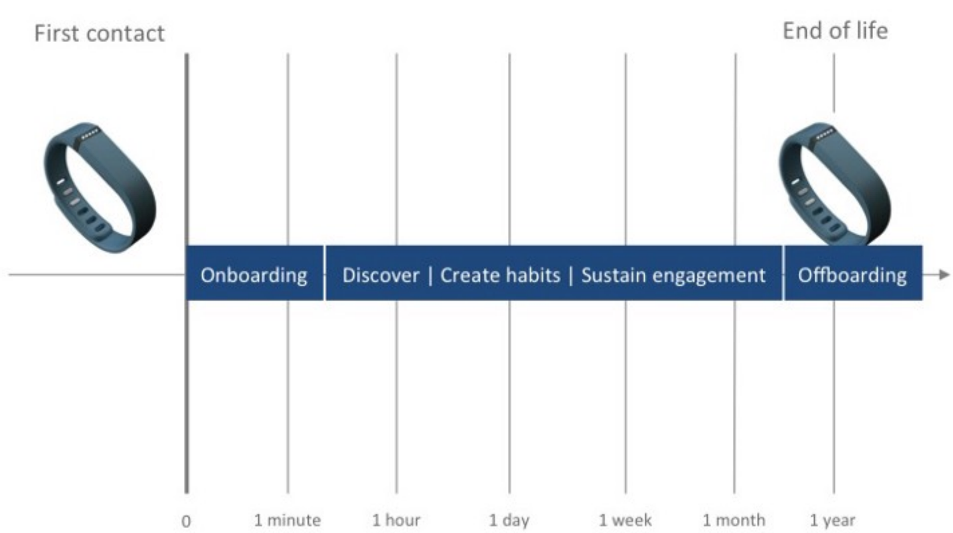

Figure 2. An experience that evolves according to behavioral data that constantly feed algorithms (e.g. Fitbit) is an experience that inevitably also has a tendency to die.

precision – metrics that often work against one another – but will also typically return some information on the precision of the information they are giving. While data scientists mostly work towards maximizing that precision, designers can use the uncertainty associated with predictions to create experiences that inform users. Designers must consider how well predictions of varying precision will support users actions, and how to exploit failures and limitations to improve, rather than hinder, an experience.

Design for engagement. Systems that learn from users behaviours are often designed to promote relevant content in order to increase engagement, particularly in domains where advertising plays a key role. Major online services are fighting to hook people and grab their attention for as long as possible: their business is to keep users active as long and frequently as possible on their platforms. This leads to the development of sticky, needy experiences that often play with users' emotions. Today, designers can use data and algorithms to exploit cognitive vulnerabilities of people in their everyday lives. In this 'attention economy,' both designers and data scientists should learn about how the experiences they craft affect the anxieties, obsessions, phobias, and stress of users. That new power raises the need for new design principles in the age of machine learning (Weyenber 2016).

There are many more experiences that go beyond this initial list. They include, for example, designing for *time well spent* (does the service aim to promote relevance, speed, and timeliness, as opposed to always-on engagement), for *peace of mind* (experiences that promote safety by detecting and explain abnormal situations).

Common Patterns

Within any single product that learns from its user behaviors, we believe there are roughly three *different* experiences to design:

1. How will the user be onboarded and familiarized with the product, particularly one that knows nothing (yet) about this user? For example, in many systems across the web, services would like to recommend interesting things (movies, music, restaurants, or travel destinations). How do you recommend something to a user that you have never seen before? Researchers call this the *cold start* problem: systems cannot draw any useful inferences about a user before gathering some information about them. This often leads to a vicious cycle: it is hard to gather useful data about users without engaging them, and it is hard to engage them without having some data to enhance their experience. This stage is therefore characterized as being both one that delights and attracts a user into a product, as well as one that draws out essential data to bootstrap any learning algorithm.
2. How will the experience evolve as the system learns; does it improve, or does it change? Machine learning algorithms are designed to look for patterns within a set of sample behaviors to probabilistically estimate the rules underlying these behaviors. This approach comes with a certain degree imprecision and unpredictable behaviors. Consequently, they require responsible design that considers moments when things start to disappoint, embarrass, annoy, stop working or stop being useful. This stage is, therefore, about finding a user experience that balances between the global predictive power of machine learning and the edge-cases that, in practice, can disassemble the value that users are getting from the product.
3. Finally, how does the experience end? The design of the 'offboarding' experience could become almost as important as the 'onboarding' one. For instance, allegedly a third of the Fitbit users stop wearing the device within 6 months. What happens to these millions of abandoned objects? What happens to the data and intelligence on the individual that they produced? What are the opportunities to use them in different experiences?

Multidisciplinary Experience Design

Traditionally, designers define the experience of a service, product or feature. They conduct qualitative research in order to identify pain points, tasks that users are trying to complete, and to formulate how a feature's goal nests into the organization's ecosystem—mapping out a user's experience and touch points within the product. They then brief data scientists and engineers to develop the algorithms that are required to support that experience.

Within data-driven organizations, teams with data scientists are changing that dynamic: they now partner directly with engineers, designers, and product managers and are involved throughout the entire product lifecycle. There are a number of reasons for this shift. First, there are many opportunities to create features that arise from the *current availability of data*, rather than a specific user need. Indeed, the research literature is peppered with papers that describe potential applications that use data that was created as a byproduct of an altogether different product. In this case, the ability to map between an available data set (for example, geo-located user interactions) and a well-known machine learning application (e.g., learning to rank) is critical at the early stages of identifying what could be built. Similarly, it is useful to consider the effects of designing user interactions (such as clicks and transactions) as artifacts *that produce data*—namely, how will this data feed back in to improve the user experience? Finally, when systems are designed to produce outputs that depend on the unique data of many individuals, there is no way to validate their performance on a case-by-case basis. Conducting and analyzing the results of large-scale online A/B tests is now a hallmark means of making design decisions based on behavioral evidence from users.

In effect, the role of the data scientist is becoming more proactive, to the point of redefining the current human-centered design approach. However, as with all multidisciplinary endeavors, we have noticed that the partnership between designers and data scientists must overcome a lack of shared understanding of each other's practice and objectives.

Design Processes and the Scientific Method

Data scientists, instead, create knowledge that can drive a user's experience using data and machine learning. This process has a strong dependence on well-formulated research questions, that are used define the hypotheses, metrics, and suitable models to develop for a given context. They follow the scientific method, which is a relatively strict, cyclical processes of constant refinement; often, this process entails jumping back and forth between tasks that centre on exploratory, analytic outcomes, and others that focus on implementation and deployment (Figure 3). This process becomes particularly visible when evaluating a proposed machine learning model. Algorithms are subject to two phases of evaluation: they get evaluated before they are deployed to production, compared to a given baseline (either the existing system, or, for example, a simple approach that relies on counting data instead of using machine learning (Zinkevich, 2017)), and they get evaluated after being deployed to production, via online A/B tests. In practice, the output of a machine learning model never has any inherent value; it is only valuable compared to an alternative.

Data scientists employ processes similar to human-center design but are more mechanical and less organic. The scientific method is similar to any design approach that forms and makes new appreciations as new iterations are necessary. Yet, it is not an open-ended process. It has a clear start and end but no definite timeline.

User Models in Design and in Data

Another area where we have noted a significant difference between designers and data scientists is when it comes to *understanding* the user. Research in the social and psychological sciences, and qualitative research for new products,

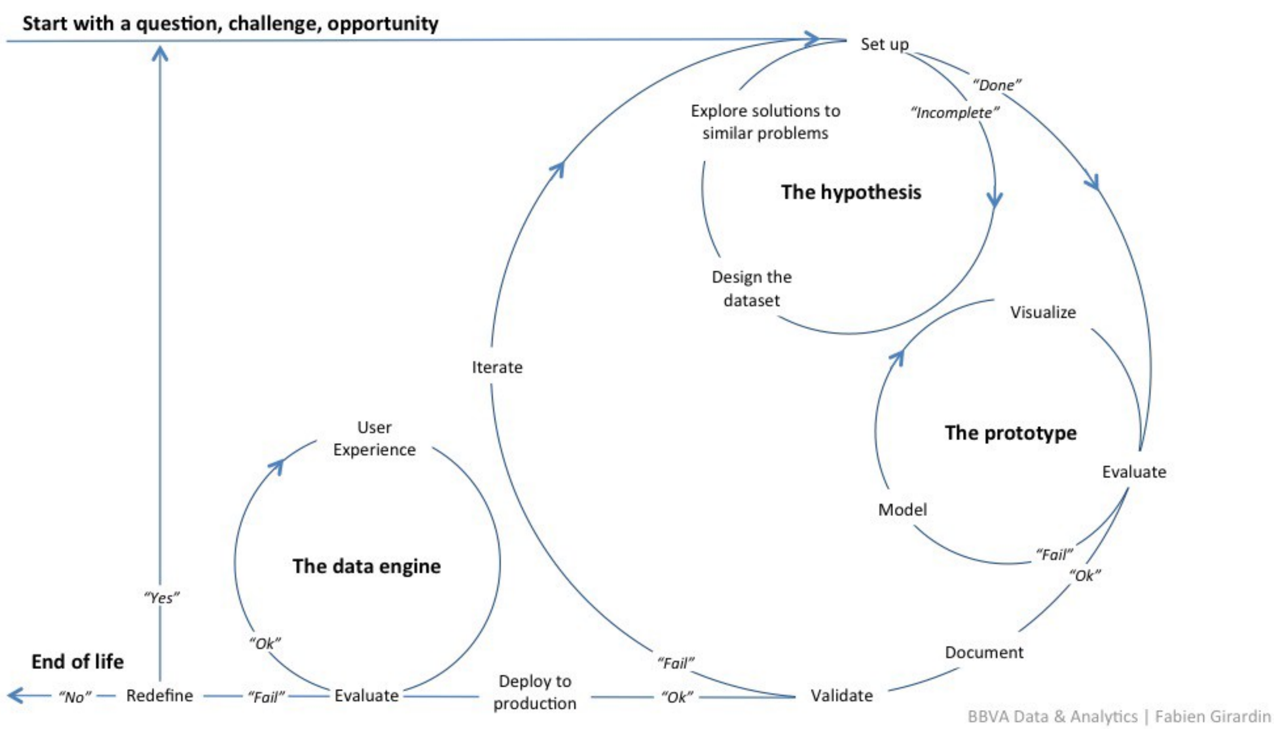

Figure 3. The data science method and its cyclical processes of constant evaluation and refinement. A properly formulated research question helps define the hypothesis and the types of models to develop in the prototyping phase. The models are the algorithms that get evaluated before they are deployed to production into a "data engine". Whenever the experience supported by the "data engine" does not perform as expected, the problem needs to be reformulated to continue the cyclical process of constant refinement.

often aims to produce *conceptual model*s about the types of people in a particular domain (for example, target users or customers of varying types). Designers will use conceptual models as a framework to brainstorm, discuss, and iterate on products that could fit with these stereotypical users--they support building a narrative around who we are building for, and what value we are delivering to them.

Often, *none* of these insights will be fundamentally encoded into the *statistical models* that are used in machine learning to deliver value to one of these users. For example, collaborative filtering algorithms are not hard-coded to comply with any specific findings from decades of research, by psychologists and behavioral economists, into how users make choices: if they did, they would not be *statistical* approaches anymore. Machine learning is, instead, evaluated according to performance metrics, rather than a deep understanding of what it is actually doing with the data under the hood**.**

The analogy between psychological research into choice overload and collaborative filtering algorithms is not meant to highlight a mismatch between the two fields. Conceptual models highlight that users are 'bad' at making choices—they motivated the need for building recommender systems in the first place. This conceptual model was then *translated* into a high level question: could data help us navigate choice? That question was then *translated* into a more specific question: how can I identify the items in a database that a user would be most interested in? Finally, this process of translation reaches something that can be turned into a statistical model: how can we rank content in a database based on signals of preference? In effect, bridging between the two fields is a process of translation.

Conclusions

In this article, we have argued that, with the advance of machine learning, it becomes the responsibility of both designers and data scientists to understand how to shape experiences that improve lives. That type of design of system behavior represents an evolution of human-centered design, that is now being developed across various institutions.

For that multidisciplinary practice to evolve, we believe that designers and data scientists must immerse themselves in the other's approaches to build a common rhythm. So far, we have codified several important touch points for designers and data scientists to produce a meaningful user experience powered by algorithms. They must:

1. Co-create a tangible vision of the experience and solution with priorities, goals and scope
2. Assess any assumption with insights from quantitative exploration, desk research and field research.
3. Articulate the key questions from the vision and the research. Are both sides of the team asking (and answering) questions that will contribute to the other disciplines?
4. Understand all the limitations of the data model that gives answers, as well as the limitations of qualitative methods that are informing the experience design.
5. Specify the success metrics for a desirable experience and define them *before* the release of any test to evaluate the impact of the data engine on the user experience.

Based on our experience in creating meaningful experiences with machine learning, we can articulate the following characteristics for the partnership between designers and data scientists to consider:

Feedback Loop: Data is the lifeblood of the user experience with systems that learn. The experience design must guarantee that systems are, from the outset, constructed properly fed with carefully crafted feedback loop mechanisms. Machine learning brings imperfections to the surface as part of the experience. For example, predicting is not the same as informing and a designer must consider how well the level of uncertainty in a prediction could support a user action.

Human-Machine Relationship: The combination of data and learning algorithms can trigger an evolution of multiple experiences. The user experience becomes a relationship between humans and the *machine that learns*, creating habits aligned with people's interests, finding the known unknown, discovering the unknown unknowns, communicating a certain peace of mind, or valuing time well spent. Additionally, the system should contemplate the "offboarding experience" for moments in the relationship when things start to disappoint, embarrass, annoy or stop working or being useful.

Language and Workflow: Cross-disciplinary work is overwhelmingly shaped by the abstract task of communicating the values of your discipline to others, in order to, together, build a great product. Everyone is likely to be using common words, such as 'data,' 'model,' 'segment,' and 'trend'—but referring to *very* different things; finding a way to communicate and work together is as important as getting the work done itself.

References

Levy, S. 2016. How Google is Remaking Itself as a "Machine Learning First" Company, https://backchannel.com/how-google-is-remaking-itself-as-a-machine-learning-first-company-ada63defcb70#.skextkifd

Hebron, P. 2016. Machine Learning for Designers, https://www.oreilly.com/learning/machine-learning-for-designers

Weyenber, A. 2016.The Ethics of Good Design: A Principle for the Connected Age. https://medium.com/swlh/dieter-rams-ten-principles-for-good-design-the-1st-amendment-4e73111a18e4#.ozflfzvsc

Zinkevich, M. 2017. Rules of Machine Learning: Best Practices for ML Engineering. http://martin.zinkevich.org/rules_of_ml/rules_of_ml.pdf