Measurement of Users' Experience on Online Platforms from their Behavior Logs

Deepali Jain[1], Atanu R Sinha[1], Deepali Gupta[2], Nikhil Sheoran[3], Sopan Khosla[1]

[1]Adobe Research
[2]Indian Institute of Technology Delhi
[3]Indian Institute of Technology Roorkee

Abstract. Delivering effective, personalized user experience by online platforms calls for individualized measure. Explicit measurement of experience, as mostly practiced, takes the form of satisfaction scores obtained by asking questions to users. Obtaining response from every user is not feasible, the responses are conditioned on the questions, and provide only a snapshot, while experience is a journey. Instead, we measure experience values implicitly from users' click actions (events), thereby measuring for every user and for every event. The latent experience values are obtained without-asking-questions, by combining a recurrent neural network (RNN) with value elicitation from event-sequence. The platform environment is modeled using an RNN, recognizing that a user's sequence of actions has a temporal dependence structure. We then propose eliciting value of a user's experience as a latent construct in this environment. We offer two methods: one based on rules crafted from marketing and consumer behavior theories, and another data-driven approach where we apply fixed point iteration, similar to the approach used in model-based reinforcement learning. Evaluation and comparison with baseline show that experience values by themselves provide a good basis for predicting conversion behavior, without feature engineering.

1 Introduction

Customer experience is a central function for firms and the new "battleground" for competition [2]. Firms are emphasizing experience-measurement as an imperative to benchmark actions and enhance user experience (UX)[2]. Also, personalization of UX is pervading firms' aspirations. All this calls for individual level measurement of UX. Yet, there remains considerable reliance on explicit measurement through surveys, which are prone to large non-response rates (the gold standard ACSI reports upwards of 85 percent [1]), response biases, are retrospective in nature and utilized at an aggregate level. While implicit measurement has been proposed in search satisfaction[19] to improve search, it has not been embraced in customer experience domains. Notably, experience is distinct from satisfaction. The latter is an outcome while experience is a journey[13]. In addressing these gaps between needs and current offerings we propose methods for computing UX values, that captures the process of experience as a journey.

The term UX covers experiences derived from any usage scenarios such as using knowledge software, or, using website / app for eCommerce, and includes *customer experience*. Experience is latent in the mind of users and difficult to measure [13]. By asking users explicitly about their experiences, surveys supposedly avoid this difficulty. But, very few respond to surveys, responses are conditioned on questions asked and provide only a snapshot. Instead, using clickstream logs that reflect *actual* behavior we measure the *latent* user experience. We offer two methods, each with two steps. In step one, both simulate an off-line learned model of the environment. In step two, which computes UX values, they differ; whereas one method crafts rules using consumer psychology, the other uses value iteration by drawing from reinforcement learning. We rely on an RNN with long short-term memory (LSTM) [7] units for modeling the environment. This allows us to use the multidimensional and continuous historical information encoded in the LSTM cell along with the current event to characterize states. Our approach is consistent with experience being a customer journey.

We benefit from multiple streams of literature. Theory of consumer psychology examines the construct experience and its relation to satisfaction[13], and grounds our concept for measuring UX. Research in CS on implicit measurement of search satisfaction from clickstream data[19] supports our thesis, although both in substantive problem definition and methodology, we deviate from this research as discussed later. Across disciplines clickstream data are analyzed to gain insights into user behaviors by utilizing hidden Markov models (HMM) and RNN, but none examines the measurement of UX. Toward measurement of latent UX our contributions are:

1. Introducing formulations and rules based on consumer behavior theories toward computing UX values. Moreover, combining these rules with modeling dynamics of an on-line platform using an LSTM network.
2. Additionally, introducing a data-driven method without pre-defined rules, where we define experience in terms of the value of different events which elicit delayed rewards. This flexible framework allows generalizability across domains. Specifically, measuring the *value* of interaction sequences is new.
3. A novel application of state value iteration method, commonly used for solving Markov Decision Processes and in Reinforcement Learning, to the domain of click-stream data analysis.
4. Representation of partially observable states in the journey of user as the memory cell of an RNN pre-trained to predict next event.

Note that we exclude features available in clickstream data such as types of product, page content, etc. By relying only on click action sequences to measure UX values, our method has less dependence on feature engineering.

2 Related Work and Defining Experience Value

Drawing from the rich customer experience literature in Marketing and Consumer Psychology, [13] points out "what people really desire are not products but satisfying experiences". Customer experience is a process, or, a journey over time [13], which comprises three phases - pre-purchase, during, and post-purchase. Measuring UX over the whole journey could be of interest[13]. UX comprises a "customer's behavioral, emotional, cognitive, and social responses to a firm's offerings ([13], pp. 71)." Thus, UX is a latent construct, in the mind of the user. Explicit measurement of UX by asking questions may capture *stated experience* on some aspects (e.g. emotion), but fails to capture actual actions. Our premise is that click action sequences observed in usage logs are crucial clues about *revealed experience*. Values of the latent construct UX can be computed for each phase based on phase-specific clickstream action sequence data. Considering the during-phase, footprints on a website or on a mobile app include click actions of filter, add to cart, for ecommerce; or, tasks performed during in-product usage of knowledge software. The actions are observed, but experiences are unobserved. By modeling these action sequences we assign values to latent experience, which drives actions observed in data.

We define experience as the value of being in a certain state of the environment in terms of proximity to a goal state. Consumer behavior literature highlights goal-orientation in online behavior and how goal-directed activities can achieve compelling experiences [17]. For the use case of e-Commerce, *Purchase* is an indicator of experience and consistent with goal attainment. Hence, we treat states in which *Purchase* event takes place as goal states. With a goal of making purchase, users go through several events on a site and incur transaction costs in search, time and psychological costs, which increase with efforts [15]. The events can be sequenced with respect to a goal; e.g., a sequence (*browsing, deliberate search, add to cart, purchase*). Moving forward from one stage to another in the sequence brings users closer to the goal and goal-gradient decreases [10], improving UX and encouraging behavior toward goal completion. Moreover, the process of purchase decision making itself contributes to experience [6]. From [16] we know the higher order event of *directed-buying sessions* has the highest conversion rate (12.94%), followed by the lower order stage of *search sessions* (8.02%).

The search literature in CS studies implicit measurement of satisfaction in order to improve search outcomes and finds that implicit measurement correlates with explicit, question based measures of satisfaction [8, 19]. This provides support for our thesis. Deviating from metrics such as dwell time, search results click, [19] offers a latent structural learning model of search satisfaction, which recognizes action level dependencies and uses *rich structured features*. Other efforts examine *struggling* in search to obtain relevant information [18]. The problem we study is about decision making (e.g., whether to purchase) based on online platform interactions and sets our work apart from that of search which is about obtaining relevant information. Typically, a poor search has less consequences for a user than a poor purchase, making the goal orientation stronger in our context of browsing experience. In browsing there is a hierarchical structure imposed by the site, whereas in search a poor result leads to user formulating another query which may not have a hierarchical basis. Finally, our model does not rely on features unlike these papers in search. Clickstream mining for measuring UX has been used to provide visualizations of common paths for site visitors [14] and to infer personas of users [22], but none suggests a method to extricate UX metrics from user logs, which we do.

We draw upon the literature in use of RNN to understand consumer behavior from clickstream data. Usefulness of RNN to link individual click actions to predictions is shown in [12]. For improved purchase prediction [11] depict the benefit of using sequential input of tweets for RNN. A manifestation of RNN [23] is in predicting sequential clicks for sponsored search. None of these papers investigates experience, which is a continuous evolution from sequential behaviors [13]. Traditionally, HMMs are used to model latent states for obtaining insights into user behaviors [3, 20, 4]. Our RNN model of the environment is Markovian, but in histories of states [21] as described later. The multidimensional and continuous historical information encoded in the LSTM cell is a major departure from the finite, discrete values for HMM. Previous application of Markov Chain model to clickstream includes mapping of journals based on logs available in scholarly portal [5], but does not include decision-making which we do. In the class of sequential data modeling techniques we have not seen in the literature any existing method that specifically measures the *value* of an interaction sequence. In this regard, our data-driven approach of using value-iteration has been derived from classical literature in reinforcement learning and decision theory.

3 Framework

We model the browsing behavior of on-line users of an eCommerce Website as a first-order Markov process. Consider a state space, $\mathcal{S} = \{s_1, s_2, s_3, ...\}$ and a reward function $r : \mathcal{S} \rightarrow \mathbb{R}$. At time t, a user in state $S_t \in \mathcal{S}$ receives a reward $r(S_t)$. The transition probability function is $\mathcal{P}(s_i, s_j) = Pr(S_{t+1} = s_j | S_t = s_i)$. Let the sequence of events observed in a user's browsing journey till time t be $E_1, E_2, ..., E_t$ where $E_i \in \mathcal{E} = \{e_1, e_2, ..., e_{|\mathcal{E}|}\}$. Events can be actions or sets of actions. Let a vector $\boldsymbol{H}_{t-1}$ of d dimensions encode all the historical information from the sequence $E_1, E_2, ..., E_{t-1}$. Then, the state at t is represented as a tuple, $S_t = (\boldsymbol{H}_{t-1}, E_t)$. Consider the encoding function, $g : \mathcal{S} \rightarrow \mathbb{R}^d$ such that, $\boldsymbol{H}_0 = \mathbf{0}$ and $\boldsymbol{H}_t = g(\boldsymbol{H}_{t-1}, E_t)$. Also, let us define the operator $\oplus$ such that,

$$
\begin{aligned}
S_t \oplus E_{t+1} &= S_{t+1} \\
(\boldsymbol{H}_{t-1}, E_t) \oplus E_{t+1} &= (\boldsymbol{H}_t, E_{t+1}) \\
(\boldsymbol{H}_{t-1}, E_t) \oplus E_{t+1} &= (g(\boldsymbol{H}_{t-1}, E_t), E_{t+1})
\end{aligned}
\tag{1}
$$

4 Learning Experience Values

We first build a model to simulate the dynamics of the environment and then apply two alternative methods for exploiting the learned model to extract latent experience values. The first method is based on predefined rules that experience values must satisfy. The second method is based on value-iteration, is data-driven and autonomous.

The environment is simulated using an RNN trained to predict the next event in the customer journey. The network encodes the information from the historical sequence of events in its d dimensional cell state. The gates of the LSTM unit of the RNN model the history encoding function g introduced above. The network estimates the transition probability function ($\hat{\mathcal{P}} \mathrel{\hat{=}} \mathcal{P}$) of the underlying Markov process of the environment. For every input sequence of events, a one step ahead sequence is predicted. The architecture of the model is as follows:

- **Input Layer:** The data are input in the form of sequences of events.
- **Embedding Layer:** The categorical variable, i.e. the event is then embedded into a latent space of dimension 150.
- **LSTM Layer:** The input is then fed into an LSTM layer with 200 hidden dimensions. The LSTM layer acts as the memory unit of the model. The hidden state of the LSTM is carried over as input to the future timestep, thus allowing the model to encode historical information.
- **Fully Connected Output Layer:** The output from the LSTM layer goes to a fully connected dense layer which produces the output of size $|\mathcal{E}|$ through softmax activation at each time-step of the sequence. The output at each time-step is a probability distribution vector over all possible next events.

The model is trained to minimize the categorical cross-entropy loss using Adam [9] optimization algorithm.

4.1 Rule-based method

For this first method we formalize the concepts of event base values (B) and event transition importance (TI). Then we outline intuitive rules that experience values ought to satisfy. While these rules are crafted from domain knowledge, some companies may prefer to impose own rules which conform to their specific situation. Later we show how the values B and TI along with the next event prediction model are used to compute final experience values (XV) at each state.

Drawing upon consumer behavior theories, a base value $B(e)$, is assigned to every event e, in the order of progression toward the goal task (*Purchase*, in this case). For example, a user who has added a product to cart is closer to completing the purchase-goal task than someone exploring products. Thus, we assign higher base value to the *Add to cart* event than the *Browsing* event.

An importance value, $TI(e_i, e_j)$ is assigned to a transition from any event e_i to another event e_j. This importance value captures how discriminative a transition is across purchase and non-purchase journeys. In other words, if a transition occurs equally frequently in both purchase as well as non-purchase journeys, then it is less important than a transition whose frequencies are unequal. Intuitively for example, transition from *Hedonic Browsing* to *Directed Search* is less important than that from *Directed Search* to *Add to Cart*, since the former likely occurs about as frequently in purchase and non-purchase journeys, while the latter occurs more frequently in purchase journey but less frequently in non-purchase journey. More formally, let

$$\begin{aligned} p &= \sum_{k=1}^{K}\sum_{t=1}^{\tau}(E_t^k = e_i)\wedge(E_{t+1}^k = e_j) \wedge (E_\tau^k = Purchase) \\ np &= \sum_{k=1}^{K}\sum_{t=1}^{\tau}(E_t^k = e_i)\wedge(E_{t+1}^k = e_j) \wedge (E_\tau^k \neq Purchase) \\ &\text{Then,} \quad TI(e_i, e_j) = \frac{|p - np|}{p + np} \end{aligned} \tag{2}$$

where, K is the number of event sequences and τ is the length of each sequence.

Let $S_t = (\boldsymbol{H_{t-1}}, e_i)$ and $S_{t+1} = (\boldsymbol{H_t}, e_j)$. The following rules characterize a desired property of experience value $XV(S_t)$:

$$\begin{aligned} &\text{if} \quad XV(S_t) \geq B(e_i) \quad \text{then} \quad \mathbb{E}(B(e_j)) \geq B(e_i) \quad \text{and} \\ &\text{if} \quad XV(S_t) < B(e_i) \quad \text{then} \quad \mathbb{E}(B(e_j)) < B(e_i) \end{aligned} \tag{3}$$

These rules imply that a user who is having a better experience than that indicated by the base value of the current event, is expected to transition to an event with higher base value and vice-versa. The objective is to find experience values that minimize the number of rules violated for a journey.

We propose alternative formulations for computing $XV(S_t)$. Later, we provide intuition for these formulations.

$$\Delta B_{i,j} = \omega_j \hat{\mathcal{P}}(S_t, S_t \oplus e_j)(B(e_j) - B(e_i)) \tag{4}$$

$$\begin{aligned} &\textbf{Formulation 1:} \quad XV(S_t) = \omega_0 B(e_i) + \sum_{j=1}^{|\mathcal{E}|} \Delta B_{ij} \\ &\textbf{Formulation 2:} \quad XV(S_t) = \omega_0 B(e_i) + \sum_{j=1}^{|\mathcal{E}|} TI(e_i, e_j)\Delta B_{ij} \\ &\textbf{Formulation 3:} \quad XV(S_t) = \omega_0 B(e_i) + T_z(S_t)\sum_{j=1}^{|\mathcal{E}|} TI(e_i, e_j)\Delta B_{ij} \\ &\qquad\qquad \text{where,} \quad T_z(S_t) = \frac{T(S_t) - mean(T(e_i))}{std(T(e_i))} \end{aligned} \tag{5}$$

where, $W = \{\omega_0, \omega_1, ..., \omega_{|\mathcal{E}|}\}$ is a set of unknown parameters and $T(.)$ is the time spent in a state or event. To examine each of the proposed formulations in a simple manner, consider the special case when $\omega_i = 1 \quad \forall i$. In Formulation 1, XV is defined as weighted sum of the current base value and the expected change in base values from current to next time step (equivalently, the expected base value of the next event). In Formulation 2, the importance of the transition to next event is also taken into account. Formulation 3 builds upon Formulation

2 through the incremental inclusion of the effect of normalized time spent in the current state ($T_z(S_t)$). This recognizes that time spent may impact experience. We estimate the optimal value for W by linear regression with the loss function as follows

$$
\begin{aligned}
\hat{y}_t = \sigma(XV(S_t) - B(e_i)) \quad \text{and} \quad y_t = \sigma(B(e_j) - B(e_i)) \\
\mathcal{L}_W = \sum_{k=1}^{K} \sum_{t=1}^{\tau} (\hat{y}_t^k - y_t^k)^2
\end{aligned}
\tag{6}
$$

where, K is the number of event sequences, τ is the length of each sequence and σ is a Sigmoid function with a high slope to simulate a unit step function. This is an implementation of number of rules violated in a differentiable form to facilitate gradient descent based parameter estimation.

4.2 Value iteration method

Our second method overcomes the deficiencies of hand-crafted rules which may not generalize to all domains. Herein, we need to use very little domain knowledge in the form of a reward function, r as follows

$$
r(S_t) = \begin{cases} 1, & \text{if } E_t = \textit{Purchase} \\ -\epsilon, & \text{otherwise} \end{cases} \tag{7}
$$

where, $-\epsilon$ is a small penalty. Now, consider a user traversing the state space of the environment and assimilating rewards along the way according to the above reward function. She achieves high reward in *Purchase* event and a small penalty (ϵ) everywhere else. We can now define the experience value of any state, S_t as the total expected discounted reward after t.

$$
XV(S_t) = \mathbb{E}(r(S_{t+1}) + \gamma r(S_{t+2}) + \gamma^2 r(S_{t+3}) + ...) \tag{8}
$$

where, $\gamma \in (0, 1)$ is the discounting factor. The above expression can be written in the form of a Bellman Equation as follows

$$
\begin{aligned}
XV(S_t) &= \mathbb{E}(r(S_{t+1}) + \gamma XV(S_{t+1})) \\
XV(S_t) &= \sum_{i=1}^{|\mathcal{E}|} \hat{\mathcal{P}}(S_t, S_t \oplus e_i)(r(S_{t+1}) + \gamma XV(S_{t+1}))
\end{aligned}
\tag{9}
$$

Since the state space is very large (all possible sequences of events), it is not feasible to get exact solution to this equation through methods such as dynamic programming or linear regression. To deal with this problem, we rely on a function approximation method. We define a simple linear estimation function f_θ with a set of parameters θ, to model the experience values.

$$
f_\theta(S_t) = \hat{XV}(S_t) \mathrel{\hat{=}} XV(S_t) \tag{10}
$$

We use the fixed-point iteration method to find θ. Start with random initial values, θ_0. At iteration number n, experience values for all observed states in the training data are estimated using θ^{n-1}. Based on these estimates, expected values, XV^n are calculated using the Bellman Equation.

$$XV^n(S_t) = \sum_{i=1}^{|\mathcal{E}|} \hat{\mathcal{P}}(S_t, S_t \oplus e_i)(r(S_{t+1}) + \gamma \hat{XV}^{n-1}(S_{t+1})) \tag{11}$$

The mean square error, $\mathcal{L}_\theta^n$ between expected $(XV^n(S_t))$ and estimated $(\hat{XV}^n(S_t) = f_\theta(S_t))$ values is used to update θ with gradient descent method until convergence. For a training dataset with K sequences with τ time-steps each,

$$\begin{gathered} \mathcal{L}_\theta^n = \sum_{k=1}^{K} \sum_{t=1}^{\tau} (f_\theta(S_t^k) - XV^n(S_t^k))^2 \\ \theta^n = \theta^{n-1} + \alpha \frac{d\mathcal{L}_\theta^n}{d\theta} \end{gathered} \tag{12}$$

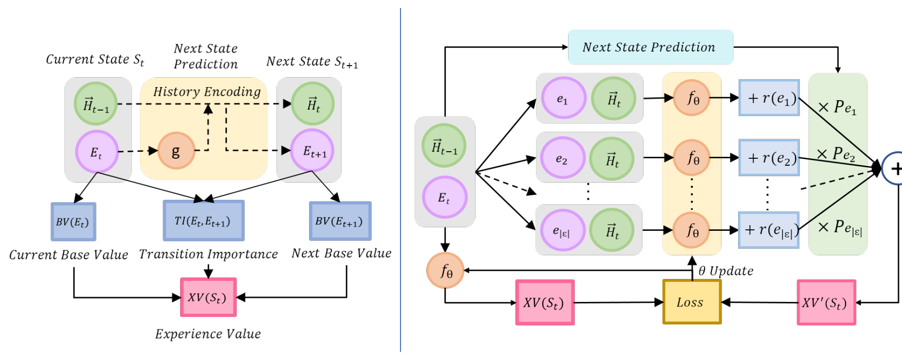

Fig. 1: Illustration of rule based (left) and value iteration (right) method

Category	Actions
Hedonic browsing (c_1)	Search, Search Filters, Product Details, Product Categories
Deliberate Search (c_2)	Reading Reviews, Product Comparison
Add to Cart (c_3)	Add to Cart, Add to List
Purchase (c_4)	Checkout, Payment, Place Order

Table 1: Actions corresponding to each category

No Pur. / Pur.	c_1	c_2	c_3	c_4
c_1	-	7153 / 2331	4075 / 8236	1694 / 4058
c_2	6626 / 2177	-	177 / 324	197 / 500
c_3	2302 / 3850	113 / 113	-	1903 / 5241
c_4	3375 / 4664	261 / 410	207 / 567	-

Table 2: Category level transition frequency for sequences (Read from category in row to category in column)

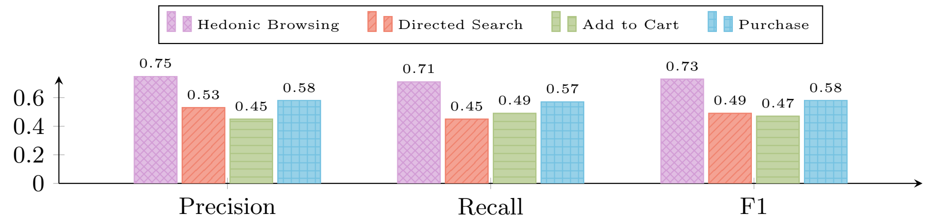

Fig. 2: Results for the next category prediction model

Method	Accuracy (%)	Precision	Recall	F1-Score	AUC
Category Sequences	66.52	0.63	**0.75**	**0.69**	**0.67**
Rule-based Form. 1 ($W = 1$)	66.64	0.69	0.65	0.67	0.66
Rule-based Form. 2 ($W = 1$)	66.48	0.71	0.64	0.67	0.66
Rule-based Form. 3 ($W = 1$)	66.64	0.73	0.64	0.68	**0.67**
Rule-based Form. 1 (optimal W)	**66.76**	0.58	0.69	0.63	**0.67**
Value iteration	63.16	**0.82**	0.59	**0.69**	0.65

Table 3: Evaluation with purchase prediction

5 Experimentation

Click-stream data from an e-Commerce site, spanning a period of three months, are used. After cleaning the data only click actions corresponding to the *Appliances* category are retained. All click actions, for each user, are stitched together chronologically into a sequence of click actions. Altogether 31 relevant click actions such as *View product details, Apply search filter and Add to cart.* are identified from the data. The set of unique actions is denoted $\mathcal{A} = \{a_1, a_2, ..., a_{31}\}$. As reasoned earlier, inspired by [16], click actions are categorized into a set of four categories i.e. *Hedonic Browsing, Directed Search, Add to Cart* and *Purchase*. Each category characterizes a different stage in a user's journey towards the goal state of purchase. The set of categories is denoted $\mathcal{C} = \{c_1, c_2, ..., c_4\}$. Categories and corresponding sample click actions are shown in Table 1. The algorithm for finding experience values is applied at category level, i.e. set of events $\mathcal{E}$ refer to the set of categories, $\mathcal{C}$. The final data are sets of sequences of events. In Table 2 we show the frequencies of transition among categories when journeys end in no purchase vs. end in purchase. Finally, the data are randomly split into two sets, training and testing, with a total of 12800 and 4600 sequences, respectively.

6 Results and Discussion

We have no access to survey based experience measurement scores of users whose usage logs we model. Firms do not share such scores. This obstacle of survey and the current use case of *goal fulfillment toward purchase* guide our evaluation. We compute a UX value for each user, for each event from usage log and then based solely on UX values predict the goal fulfillment (purchase), under the thesis that UX affects goal fulfillment. Purchase prediction is not the focus, but merely a way of evaluating the worth of derived UX values. We show that using UX

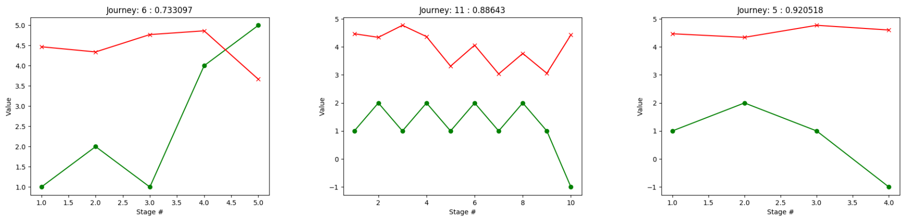

Fig. 3: Evolution of experience during journeys

values can give purchase prediction comparable to that of feature-engineered model. As exemplar of the latter, within the data we have, category (event) sequence based model is comprehensive since it captures the sequence along with frequency of sub-events and time spent on events and forms the baseline. True to our objectives, the methods are to be judged by how closely the accuracy of feature-based model can be reproduced by our methods.

A multi-layer RNN module performs purchase prediction by taking as input either sequence of events or experience values generated from one of the proposed methods. Prediction accuracies of models with fixed architecture and different inputs is then compared. The model architecture is similar to next event prediction model, with a difference in the final layer which produces a single output (purchase probability) per time-step through sigmoid activation. The training is done to minimize binary cross-entropy loss.

We evaluate both environment simulation and UX value generation models. Results for the former for next event prediction on test data are shown in Fig. 2. These measures are obtained by averaging across categories from which arrival into a category can occur. We find some variability in these measures across the categories. Results for UX value generation are shown (in Table 3) for four variations of UX value computation using the rule based method - Formulations 1-3 (AUC = 0.66, 0.66 and 0.67 respectively) and one with parameter tuning for Formulation 1 (AUC = 0.67). The results from the value iteration method (AUC = 0.65) are also compared. For each of these, purchase prediction is carried out by using the generated UX values as the only input. The baseline used is an event-sequence based prediction. We find that although the UX values are extracted based on rules, their performance in predicting purchase is very close to the baseline (AUC = 0.67), which uses features such as frequency of actions and time spent within each category. This suggests that computed experience values capture latent components of browsing experience, which explain purchase propensity as accurately as using information in raw data.

Fig. 3 depicts UX values (red), as users move through states (green), for three users. Note the red UX values are leading indicators. E.g., from state 1 to state 2 if UX value decreases, it is expected the user moves from the stage in state 2 to a lower stage in state 3. For red lines, last segments are not interpretable. The left figure shows an upward drift consistent with higher stage attainment, and early sign of UX leading to higher stage. The leading indicator between states 2 and 3 suggests movement from stage 1 to stage 4 going from state 3 to state 4. The middle figure shows a user oscillates between stages 1 and 2 over states, without ever going to a higher stage. The overall downward drift is an early indicator of poor experience and no purchase. The right figure is less informative since the slightly upward tendency is not consistent with stage traversal.

7 Conclusion

We show that UX values can be uncovered from readily available user behavior logs. Drawing from theory we grouped actions into categories to build the model. An alternative thought could be to build a model directly from the raw actions. The action-level model using value iteration shows that for the task of purchase prediction we obtain accuracy (0.67), precision (0.87), recall (0.67), F1 (0.75) and AUC (0.57). Comparing with stage-level results from the last line of Table 3 we find that in AUC, the stage level model performs better.

Rules based method may fit customers who 'live' click to click or are myopic, while value iteration captures long-view customers' behaviors. Several challenges include how to do a direct evaluation based on experience metrics obtained in a direct way. Limitations also pertain to the generalizability of the approach to non-discretionary and low involvement products. The appliance category used here constitutes discretionary spending and a high ticket purchase engendering extensive browsing behaviors. Our use case is for the during phase of the whole customer journey. With data from the pre and post phases, future work can extend the approach to mine UX values for those phases. It is noted that our approach can ingest any goal, not just purchase. For example, information seeking. As well, other rewards and intermediate rewards can be provided. None of these applies in a purchase prediction model.

References

1. The american customer satisfaction index. http://www.theacsi.org/.
2. Cx index - forrester. https://go.forrester.com/data/cx-index/.
3. C. R. Anderson, P. Domingos, and D. S. Weld. Relational markov models and their application to adaptive web navigation. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 143–152. ACM, 2002.
4. G. Balakrishnan and D. Coetzee. Predicting student retention in massive open online courses using hidden markov models. *Electrical Engineering and Computer Sciences University of California at Berkeley*, 2013.
5. J. Bollen, H. Van de Sompel, A. Hagberg, L. Bettencourt, R. Chute, M. A. Rodriguez, and L. Balakireva. Clickstream data yields high-resolution maps of science. *PloS one*, 4(3):e4803, 2009.
6. M. Chylinski, A. Sinha, D. S. Lie, and W. Neill. Experience infusion: How to improve customer experience with incidental activities. *Marketing Science Institute Working Paper Series*, (17-106), 2017.

7. S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
8. Y. Kim, A. Hassan, R. W. White, and I. Zitouni. Modeling dwell time to predict click-level satisfaction. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*, WSDM '14, pages 193–202, New York, NY, USA, 2014. ACM.
9. D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
10. R. Kivetz, O. Urminsky, and Y. Zheng. The goal-gradient hypothesis resurrected: Purchase acceleration, illusionary goal progress, and customer retention. *Journal of Marketing Research*, 43(1):39–58, 2006.
11. Korpusik, Mandy, and et al. Recurrent neural networks for customer purchase prediction on twitter. *ACM Conference on Recommender Systems*, 2016.
12. T. Lang and M. Rettenmeier. Understanding consumer behavior with recurrent neural networks. *International Workshop on Machine Learning Methods for Recommender Systems*, 2017.
13. K. N. Lemon and P. C. Verhoef. Understanding customer experience throughout the customer journey. *Journal of Marketing*, 80(6):69–96, 2016.
14. Z. Liu, Y. Wang, M. Dontcheva, M. Hoffman, S. Walker, and A. Wilson. Patterns and sequences: Interactive exploration of clickstreams to understand common visitor paths. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):321–330, Jan. 2017.
15. S. E. Masten and O. E. Williamson. *Transaction cost economics.* Edward Elgar, Aldershot, Hants, England ; Brookfield, Vt., 1995. 2 v.
16. W. W. Moe. Buying, searching, or browsing: Differentiating between online shoppers using in-store navigational clickstream. *Journal of consumer psychology*, 13(1-2):29–39, 2003.
17. T. P. Novak, D. L. Hoffman, and A. Duhachek. The influence of goal-directed and experiential activities on online flow experiences. *Journal of Consumer Psychology*, 13(1):3 – 16, 2003.
18. D. Odijk, R. W. White, A. Hassan Awadallah, and S. T. Dumais. Struggling and success in web search. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 1551–1560. ACM, 2015.
19. H. Wang, Y. Song, M.-W. Chang, X. He, A. Hassan, and R. W. White. Modeling action-level satisfaction for search task satisfaction prediction. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 123–132. ACM, 2014.
20. A. Ypma and T. Heskes. Automatic categorization of web pages and user clustering with mixtures of hidden markov models. In *International Workshop on Mining Web Data for Discovering Usage Patterns and Profiles*, pages 35–49. Springer, 2002.
21. H. Yu, A. R. Mahmood, and R. S. Sutton. On generalized bellman equations and temporal-difference learning. In *Canadian Conference on Artificial Intelligence*, pages 3–14. Springer, 2017.
22. X. Zhang, H.-F. Brown, and A. Shankar. Data-driven personas: Constructing archetypal users with clickstreams and user telemetry. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 5350–5359, New York, NY, USA, 2016. ACM.
23. Y. Zhang, H. Dai, C. Xu, J. Feng, T. Wang, J. Bian, B. Wang, and T.-Y. Liu. Sequential click prediction for sponsored search with recurrent neural networks. In *AAAI*, pages 1369–1375, 2014.