Using Machine Learning to Address Customer Privacy Concerns: An Application with Click-stream Data

Hyesung Yoo Song Yao Luping Sun Xiaomeng Du*

This draft: January 21, 2019

*Hyesung Yoo is a PhD candidate in Marketing at the Carlson School of Management, University of Minnesota. Song Yao is an Associate Professor of Marketing at the Carlson School of Management, University of Minnesota. Luping Sun is an Associate Professor of Marketing at the Business School of Central University of Finance and Economics. Xiaomeng Du is the Chief Data Scientist at the Baifendian Information Technology Co., Ltd. Please contact Yoo (hyesung@umn.edu) or Yao (syao@umn.edu) for correspondence.

Using Machine Learning to Address Customer Privacy Concerns: An Application with Click-stream Data

Abstract: The ever-increasing volume of consumer data provide unprecedented opportunities for firms to predict consumer behavior, target customers, and provide customized service. Recent trends of more restrictive privacy regulations worldwide, however, present great challenges for firms whose business activities rely on consumer data. We address these challenges by applying the recently developed federated learning approach - a privacy-preserving machine learning approach that uses a parallelized learning algorithm to train a model locally on each individual user's device. We apply this approach to data from an online retailer and train a Gated Recurrent Unit recurrent neural network to predict each consumer's click-stream. We show the firm can predict each consumer's activities with a high level of accuracy without the need to store, access, or analyze consumer data in a centralized location, thereby protecting their sensitive information.

Keywords: Privacy, Machine Learning, Federated Learning, Gated Recurrent Unit, Click-stream data

1 Introduction

As consumers constantly generate massive amounts of data, unprecedented opportunities exist for firms to harness the power of individual-level consumer data to predict their behavior and to target and customize service to consumers. The rapid growth of the use of consumer data, however, has also increased debate surrounding the protection of consumers' privacy. The Privacy Rights Clearinghouse reports that 8,909 data-breach incidents have been made public since 2005, compromising billions of sensitive personal records.[1] The scope and the extent of data breaches are alarming. For instance, millions of users were affected by the 2017 Equifax data-breach incident that exposed sensitive personal information such as driver's license numbers, credit history, and even social security numbers.[2]

Consumers have expressed serious concerns pertaining to how firms handle consumers' data and protect their privacy. According to an online survey conducted by IBM in 2018, 78% of U.S. consumers said that a company's ability to keep consumer data private is "extremely important," and only 20% responded that they "completely trust" the companies they interact with to keep their private data safe.[3] Another survey by Consumer Reports finds that in the aftermath of Facebook's Cambridge Analytica scandal in 2018, in which the British consulting company deceitfully acquired and used millions of Facebook users' data, 70% of Facebook users have changed their behavior, taking more precautions with their posts, revising privacy settings, and turning off location tracking.[4] These examples show that with growing concerns over privacy issues, consumers have become skeptical of firms' promises about the use and protection of consumers' personal data. As a result, firms are now facing a crisis of trust and confidence from their consumers.

[1] Source: https://www.privacyrights.org/data-breaches, accessed on December 1, 2018.

[2] Source: https://arstechnica.com/information-technology/2018/05/equifax-breach-exposed-millions-of-drivers-licenses-phone-numbers-emails/, accessed on December 1, 2018.

[3] Source: http://analytics-magazine.org/survey-finds-deep-consumer-anxiety-over-data-privacy-and-security/, accessed on November 25, 2018.

[4] Source: https://www.cmswire.com/information-management/how-facebooks-cambridge-analytica-scandal-impacted-the-intersection-of-privacy-and-regulation/, accessed on November 21, 2018.

Governments are also concerned with the adequacy of data security and protection of consumer privacy implemented by companies. Accordingly, governments in many countries are considering regulations that greatly restrict firms' access, use, and sharing of consumer data. One noteworthy privacy legislation is the European Union's General Data Protection Regulation (GDPR). With the goal of creating more consistent protection of consumer personal data across all EU nations, the GDPR went into effect on May 25, 2018, as the primary law regulating how companies protect EU citizen's personal data.[5] Under GDPR, organizations must obtain explicit consent from users in order to store users' personal data, and also have a legal obligation to inform users of the purpose of data collection and processing, as well as of the identities of third parties with whom the data will be shared.[6] Companies that fail to comply with the GDPR are subject to costly penalties of up to €20m, or 4% of a firm's global turnover of the previous year (whichever is greater). Furthermore, note that in addition to EU members, any company, regardless of its location, must comply with the regulation if it markets goods and services to EU residents (known as "extra-territoriality"). The impact of the GDPR thus exceeds the boundaries of EU and changes data-protection requirements globally.

The GDPR is just the beginning - recent high-profile data breaches have further triggered calls for more urgent and strict data-protection measures worldwide. For example, modeled after the GDPR, the California Consumer Privacy Act of 2018 (CCPA) was recently passed (June 2018) and will become effective in 2020. Much like the GDPR, the CCPA provides consumers more control over their personal information by requiring California-based organizations to obtain explicit consent from users before sharing or selling consumer data to third parties. India is also one step closer to having its own data-protection law. In July 2018, the Indian government published the draft of the Personal Data Protection Bill, which

[5] According to GDPR directive, "personal data" are defined as "any information relating to an identifiable person who can be directly or indirectly identified by reference to an identifier. This definition provides for a wide range of personal identifiers to constitute personal data, including name, identification number, location data or online identifier, reflecting changes in technology and the way organizations collect information about people."

[6] Source: https://eugdpr.org/the-regulation/gdpr-faqs/, accessed on November 21, 2018.

proposes a comprehensive data-protection framework and is similar to the GDPR in terms of extra-territoriality and global-turnover-based penalties.

While offering more rights and protection to consumers, such strict regulations will inevitably limit firms' ability to tailor their marketing activities and services to each individual consumer. Not only will these regulations negatively affect the profitability of firms that rely heavily on individual consumer data for prediction and targeting, but their impact on consumer welfare is also ambiguous. Note that firms' targeting activities often provide additional value to consumers, for instance, through lower search costs or through a better match with a product (e.g., Yao and Mela (2011), Anderson and Simester (2013)). Consequently, it is unclear whether consumers will eventually be better off if firms stop exploring consumer data under these new privacy policies. Therefore, it is imperative to find solutions that can alleviate the potential negative side effects of restrictive privacy regulations, while preserving data security.

In this paper, we show how machine learning approaches can achieve such objectives by enabling firms to continue benefiting from the abundance of consumer data without the need to store or access the data, hence mitigating the privacy concerns. In particular, we demonstrate how firms may achieve accurate targeting without centralized storage or access to the data, by building a Gated Recurrent Unit (Cho et al. (2014)) recurrent neural network (RNN) under the Federated Learning algorithm (McMahan et al. (2017)). The Gated Recurrent Unit (GRU, henceforth) recurrent neural network algorithm can achieve a highly accurate prediction about a consumer's next action conditional on what she has done or experienced in the past (e.g., a firm can predict which movie a consumer is more likely to watch based on her watch history and which movies are recommended to her). The Federated Learning (FL, henceforth) algorithm stores the private data locally on each user's device, while the model parameters are also updated locally on that device using those data. During the training, the firm does not need to access the private data directly, thereby keeping them safe. Only those locally updated parameters from consumers' devices are communicated to the central server (firm). Upon receiving those updated parameters from consumers, the firm aggregates them to update a "shared" model.[7]

The FL approach has a distinct advantage over other methods devised to protect privacy. Even if mostly anonymized, datasets that are stored and accessible at the firm's data center can still put consumer privacy at risk (Sweeney (2000)). For instance, consider the Differential Privacy algorithm (Dwork et al. (2006)) that Apple has deployed since 2016 as a key feature to protect consumer identity. When Apple collects and stores user data, it adds statistical noise to a user's profile and activities to mask the user's identity. A study by Tang et al. (2017) finds, however, that Apple's privacy-breach risk still exceeds the level that the research community typically considers acceptable. By contrast, the FL trains the model on each consumer's device locally, and therefore greatly reduces such risks because the firm never transfers, accesses, or stores consumers' personal data. The only information that is transmitted between the firm and consumers is the locally updated parameters that are necessary to improve the shared model.

Another attractive property of FL, which also distinguishes it from other distributed learning algorithms, is that it is robust to non-IID and highly unbalanced datasets. The data stored on any given consumer's device are almost certainly not representative of the population distribution, and the amount of data stored will vary substantially based on the consumer's usage of the device and the firm's service. While much of the previous research on distributed learning does not consider unbalanced and non-IID datasets, the FL approach works relatively well on these types of data by repeatedly averaging locally updated parameters.

Furthermore, the FL is communication efficient. One major constraint in the design of large-scale distributed learning algorithms is the communication cost. In a typical distributed learning setting where the data are stored in a decentralized manner over a cluster of devices

[7] In the machine learning literature, "parameters" and "weights" are often used interchangeably. We use "parameters" to distinguish our meaning from "weights" used in weighted-averaging calculations, which appear later in the paper.

(nodes), communication costs are considerable. The development of an efficient distributed learning algorithm that can minimize the number of communication iterations among nodes is therefore an important issue. In the FL setting, the network and power connection of a consumer's device make communication costs the principal constraint. McMahan et al. (2017) demonstrate how two components of the FL approach can substantially reduce the number of communication rounds necessary for achieving a target accuracy level. The two components are (1) increasing parallelism, so that more consumers do computation independently during each communication round, and (2) increasing computation on each consumer's device, so that multiple updates are performed at the consumer level during each communication round.

To demonstrate the applicability of the proposed approach in a general marketing setting, we train the GRU with the FL algorithm using a highly unbalanced and non-IID consumer browsing dataset at an online retailer, with the objective to predict a consumer's clickstream. To establish a benchmark, we also train the GRU using a standard centralized learning approach. In contrast to the FL algorithm, the centralized learning requires the firm to store, access, and train all consumers' data collectively at a data center. We show the prediction accuracy of the proposed approach is comparable to that of the centralized learning method. Consequently, this approach allows firms to target consumers with high accuracy without compromising the security of personal data.

The rest of this paper is structured as follows: In the following section, we briefly discuss related literature. Section 3 gives a brief overview of the FL algorithm, as well as the GRU. In Section 4, we apply the FL algorithm with the GRU to a practical marketing problem, training a model to predict each consumer's next-clicked item using an online retailer's click-stream data. Section 5 concludes.

2 Related Literature

This paper adds to a stream of literature on consumer privacy. Hann et al. (2003) study the trade-off that consumers face between the benefits and costs of providing personal information. They find the benefits such as monetary rewards and future convenience significantly affect consumers' preferences over websites with various privacy policies. They also quantify individuals' valuation of protection of personal information, and find it is worth between \$30.49 and \$44.62. Tucker (2014) shows that increasing users' perception of more control over their private information increases the effectiveness of behavioral targeting. Leveraging the implementation of European Union's opt-in tracking policy as a natural experiment, Goldfarb and Tucker (2011) demonstrate that display advertising becomes far less effective (65% reduction in effectiveness on average) in terms of stated purchase intent as a result of the privacy regulation. In the context of the online display ad industry, Johnson (2013) finds that reduced targeting due to stricter privacy policies decreases advertiser surplus, and that publishers' revenues also decrease as a result. More recently, Rafieian and Yoganarasimhan (2018) use machine learning techniques to quantify the value of targeting information, specifically, the relative importance of contextual information (based on the content of the website and hence privacy preserving) versus behavioral information (based on user-tracking and thereby jeopardizing privacy). They find that targeting consumers based on behavioral information is more effective than targeting based on contextual information, and that strict privacy regulations that ban user-tracking substantially reduce the value of behavioral targeting. For a more comprehensive review and discussion on big data and consumer privacy, see Jin (2018).

This paper also relates to literature on privacy-preserving machine learning (Barni et al. (2011), Xie et al. (2014), Rubinstein et al. (2012), Sarwate and Chaudhuri (2013), Duchi et al. (2012), Mohassel and Zhang (2017)). Privacy-preserving deep learning has been an active research area in recent years. The most relevant study in this domain is Shokri and Shmatikov (2015), who propose a method based on Differential Privacy (DP) for collaborative deep learning, where each party asynchronously trains a neural network locally and selectively shares only a subset of parameters with other parties. They do not, however, take into account the non-IID and unbalanced properties of the data. McMahan et al. (2017) advance this literature by developing the FL algorithm that is robust to unbalanced and non-IID data distributions that are the defining characteristics of data stored in each consumer's device. This distributed learning technique offers the firm as well as consumers the benefits of the shared model trained from rich data, without having to compromise the security of personal data. In our paper, we further combine the FL approach with the GRU approach. We use the model to predict consumer click-streams and demonstrate its accuracy and applicability in marketing.

This paper also belongs to the literature that explores path-tracking and click-stream data to study consumers' decision-making along the purchase funnel (e.g., Moe and Fader (2004), Montgomery et al. (2004), Park and Fader (2004), Hui et al. (2009)). Unlike typical brick-and-mortar data, which only record consumers' final transactional events, path-tracking and click-stream data can accurately capture the entire shopping path of a consumer in a complete and timely manner. As shown in recent studies, insights obtained from such data can provide a better understanding of consumers' search behavior and market competition, as well as enable managers to optimize their marketing efforts (e.g., Bronnenberg et al. (2016), Chen and Yao (2017), Seiler and Yao (2017), Yao et al. (2017)). Tracking and storing path and click-stream information, however, also intensifies privacy concerns. Even after the data are anonymized, the empirical patterns embedded in the data can reveal a substantial amount of personal information (Valentino-DeVries et al. (2018)). Our paper demonstrates the possibility of analyzing path-tracking and click-stream data without jeopardizing consumers' privacy.

3 Model

In this section, we provide a brief description of the FL as well as the GRU algorithms. We first describe the FL's process of model distribution and aggregation executed by the central server, and then proceed to describe the GRU algorithm that trains the model locally at each individual consumer's device using personal data.

3.1 Server

The data are partitioned over K consumers, with n_k number of observations for consumer k, $k = 1, ..., K$. Let $\mathcal{P}_k = \{1, ..., i, ..., n_k\}$ be the set of indices for consumer k's data points; that is, $n_k = |\mathcal{P}_k|$. At round τ of communication between consumer devices and the central server, a fraction $C \in (0, 1]$ of all consumers are randomly selected to form a set S_τ (i.e., only a fraction C of consumers are selected during each communication round for computational efficiency). The model parameters of the current round, Θ_τ, are distributed from the central server to all consumers who have been selected to be included in this set. Next, each consumer k's device computes the average gradient g_k on her local data at the current parameters Θ_τ. The average gradient g_k can be written as $g_k(\Theta_\tau) = \nabla L_k(\Theta_\tau)$, where $L_k(\Theta_\tau) = \frac{1}{n_k} \sum_{i \in \mathcal{P}_k} l_i(\Theta_\tau)$, and $l_i(\Theta_\tau)$ is the loss function of the prediction on observation i.

The parameters are locally updated as

$$\Theta^k_{\tau+1} \leftarrow \Theta_\tau - \eta g_k, \tag{1}$$

where η is a learning rate. In other words, in parallel, each consumer locally takes one step of gradient descent at the current parameters using her local data. The resulting parameters, $\Theta^k_{\tau+1}$, $\forall k \in S_\tau$, are sent to the central server. The central server then takes a weighted average of parameters received from the consumers and updates the shared central model $\Theta_{\tau+1} \leftarrow \sum_{k \in S_\tau} \frac{n_k}{n_{S_\tau}} \Theta^k_{\tau+1}$, where n_{S_τ} is the total number of observations across all consumers in S_τ. This process repeats until convergence.

Sometimes, increasing local training epochs may further improve the communication efficiency (i.e., reduce the number of communication rounds necessary for convergence).[8] Specifically, during round τ of communication, instead of updating the local parameters only once at each consumer's device, it is possible to modify the procedure by increasing the number of local training epochs to $E > 1$ times before communicating to the central server. Let e be the index of local training epochs. Then consumer k's parameters at round τ are updated as

$$\Theta_{\tau+1}^{k,e+1} \leftarrow \Theta_{\tau+1}^{k,e} - \eta g_k(\Theta_{\tau+1}^{k,e})$$

$$e = 1, 2, ..., E$$

$$\text{with } \Theta_{\tau+1}^{k,1} = \Theta_\tau \text{ and } \Theta_{\tau+1}^{k} = \Theta_{\tau+1}^{k,E+1}.$$

The improvement in communication efficiency through this additional step, however, is not guaranteed. The improvement in efficiency may depend on characteristics of data that are stored on each consumer's device (e.g., sparsity). As we show in our application in Section 4 (as well as shown in McMahan et al. (2017)), the additional local training epochs may not necessarily enhance the speed of neural network convergence. Accordingly, in practice, firms need to fine-tune the number of local epochs to achieve a high level of communication efficiency.

3.2 Consumer k

At each consumer's node, we employ the GRU to predict each consumer's next-clicked item during a browsing session. The GRU solves the vanishing gradient problem of the vanilla RNN using an "update gate" vector and a "reset gate" vector. These two gates determine how much information from a consumer's previous clicks needs to be passed along to make

[8] In the machine learning literature, an "epoch" is defined as one round of passing *all* data forward and backward through the network. Because the training happens on each individual consumer's device and all her data are passed through the local neural network, each training iteration can be viewed as one local epoch.

predictions about future clicks. They can be trained to retain information from multiple steps back or to ignore the information that is irrelevant for the prediction. For notational simplicity, we omit the indices k and τ that index a specific consumer and a communication round, respectively.

During a specific browsing session, a consumer makes $T \geq 2$ clicks. Suppose J alternative products are available at each session. At step t ($t = 1, 2, ..., T$) of the browsing session, the consumer can choose one product to click. Let matrix $X = [x_1, x_2, ..., x_T]$ be the sequence of vectors representing the consumer's click-stream in a given browsing session. $x_t \in \mathbb{R}^{J \times 1}$ is a J-dimensional vector whose j-th element equals 1 if a consumer clicks on product j at step t, and 0 otherwise.

Given the sequence $[x_{1,}x_2..., x_t]$ up to step t, $t = 1, 2, ..., T-1$, our objective is to predict x_{t+1}, the click vector at step $t+1$. At each t, $t = 1, 2, ..., T-1$, the hidden state of the previous step, $h_{t-1} \in \mathbb{R}^{D \times 1}$,[9] and the input x_t are passed to the gated recurrent unit.[10] The gated recurrent unit in turn updates the current hidden state h_t ($t = 1, 2, ..., T-1$) using the following architecture

$$z_t = \sigma([W_z x_t + U_z h_{t-1} + b_z) \tag{2}$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{3}$$

$$\hat{h_t} = tanh(W_h x_t + U_h(h_{t-1} \odot r_t) + b_h) \tag{4}$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \hat{h}_t, \tag{5}$$

where z_t and r_t are update and reset gates, respectively; $\hat{h}_t$ and h_t are the current memory and the hidden state, respectively; $\sigma(\cdot)$ is the sigmoid function; $tanh(\cdot)$ is a hyperbolic tangent function; and $\odot$ denotes an element-wise multiplication. $W_z, W_r, W_h, U_z, U_r, U_h$ are the matrices, and b_z, b_r, b_h are the vectors of parameters to be learned. The intuition of the GRU is as follows:

[9] $D \times 1$ is the dimension of the hidden state vector.

[10] Note h_0 is a vector with all elements equal 0.

Update gate (equation 2): The update gate z_t allows the model to control how much of the information from previous steps (which is summarized in h_{t-1}) should be carried forward to the current hidden state h_t. The update gate helps the model remember long-term information.

Reset gate (equation 3): Despite their identical formula, the reset gate r_t is different from the update gate z_t. The difference comes from the parameter matrices and vectors, and more importantly, the gate's usage. The reset gate r_t allows the model to drop any previous information that is irrelevant for future predictions.

Current memory (equation 4): The current memory $\hat{h_t}$ consolidates the new input x_t (the click vector in step t) with the previous hidden state h_{t-1}. The latter holds information from the consumer's click activities in previous $t-1$ steps.

Hidden state (equation 5): The hidden state h_t uses the update gate as the weight to store relevant information from the previous hidden state h_{t-1} and the current memory $\hat{h_t}$.

The hidden state h_t is then used to calculate the prediction of the click vector of step $t+1$, $\hat{x}_{t+1}$. The prediction $\hat{x}_{t+1}$ takes the form of a J-dimensional vector, whose j-th element is the probability of the consumer clicking product j. Specifically,

$$\hat{x}_{t+1} = \left[\frac{exp(o_{t,1})}{\sum_{j=1}^{J} exp(o_{t,j})}, \cdots, \frac{exp(o_{t,J})}{\sum_{j=1}^{J} exp(o_{t,j})} \right]' \tag{6}$$

$$o_t = [o_{t,1}, o_{t,2}, ..., o_{t,J}]' \tag{7}$$

$$= Vh_t + b_v, \tag{8}$$

where V and b_v are another set of matrix and vector of parameters to be learned.

Finally, we use the cross-entropy error as the loss function, which is defined as

$$L = \frac{1}{T-1} \sum_{t=1}^{T-1} x_{t+1} \cdot log(\hat{x}_{t+1}). \tag{9}$$

The full set of model parameters to be learned are

$$\Theta_\tau = \{W_u, U_u, b_u, W_r, U_r, b_r, W_h, U_h, b_h, V, b_v\}. \tag{10}$$

4 Application: Click-stream Prediction

We apply the FL algorithm to a click-stream dataset from an online retailer, and train the GRU locally at each consumer's node using only that consumer's personal data. Our goal is to show how the FL algorithm can fit into a broad marketing framework. In particular, we combine the FL with the GRU to test the performance of the prediction of each consumer's click-stream within a browsing session. As discussed in Hidasi et al. (2016), the prediction of the next-clicked product or a set of products in a customer's click-stream often become the basis for a website's recommendation system. A well-calibrated recommendation system in turn may enhance the conversion rate of the online retailer. Consequently, accurately predicting a consumer's click-stream within a browsing session has substantial managerial implications. To evaluate the accuracy of the prediction, we focus on the predicted probability on the next-clicked product. In particular, we use "Recall@K" averaged over all clicks of all consumers as our evaluation metric of prediction accuracy. Recall@K is widely used in the machine learning literature for predicting click-through rates (Hidasi and Tikk (2016)). For our application, the consumer clicks on only *one* product at each step. In this case, Recall@K is a dummy variable. More specifically, for a given prediction at step t, Recall@K equals 1 if the list of K products with the highest predicted click probabilities includes the product that the consumer actually clicks. Recall@K equals 0 if the actually clicked product does not appear in the K-product list.

Table 1: Summary Statistics of Training Dataset

	Training Set				
	Mean	SD	Med	Min	Max
Number of sessions per customer	1.89	2.55	1	1	52
Number of clicks per customer	28.43	137.89	9	2	7,332
Number of clicks per session	15.03	49.68	5	2	1,844
Number of unique products clicked per customer	6.34	9.85	4	1	217
Number of unique products clicked per session	4.40	4.55	3	1	88
Number of customers	3,632				
Number of sessions	6,873				
Number of clicks	103,270				

Table 2: Summary Statistics of Test Dataset

	Test Set				
	Mean	SD	Med	Min	Max
Number of sessions per customer	1.85	2.12	1	1	28
Number of clicks per customer	26.93	73.44	9	2	1,212
Number of clicks per session	14.58	42.35	5	2	1,008
Number of unique products clicked per customer	6.47	8.91	4	1	125
Number of unique products clicked per session	4.39	4.23	3	1	43
Number of customers	908				
Number of sessions	1,677				
Number of clicks	24,454				

We use a dataset from a large Chinese online liquor retailer, which contains a set of 5,711 randomly selected customers shopping in the wine category on the website during July 2016. For each customer, we observe her individual-level click-stream at the website. During the observation window, these 5,711 customers initiate 13,154 browsing sessions on the website.[11] During these sessions, they make 132,328 clicks on 1,660 products.

On average, each product appears in approximately 275 sessions, but with a large variance. Some unpopular products only appear once in browsing sessions across customers, while the most popular product appears in 1,177 sessions. We aggregate unpopular products that appear in less than five sessions into one composite good. There are 798 such products, and they constitute only 2.04% of total clicks in the data. We also drop sessions in which a customer makes only one click, because our objective is to predict the next-

[11] A session ends when the customer closes the website's browser window/tab.

clicked item during the browsing session. As a result, we lose 4,604 observations (clicks) after dropping those sessions.

Our final sample consists of 127,724 clicks, 4,540 customers, 8,550 sessions, and 863 products (862 products and one composite good). On average, we have 1.89 sessions per customer, each session consists of 15 clicks, each customer has 28 clicks, and each customer clicks on 4.1 unique products per session and 6.37 unique products in total.

We randomly select 80% of unique customers for training and use the remaining 20% as a test set to calibrate the out-of-sample prediction accuracy. Our training dataset contains 103,270 clicks, 3,632 customers, and 6,873 sessions. Our testing data consist of 24,454 clicks, 908 customers, and 1,677 sessions. Note that due to our random assignment of consumers into training/test sets, the total number of unique products clicked vary across the two groups of consumers, even though they face the same set of alternative products. Summary statistics of the training set and the test set are reported in Table 1 and Table 2, respectively.

Each shopping session of a consumer forms a separate sequence. That is, if any of the consumer's sessions ends, we reset the appropriate hidden state. We fix the size of the hidden states to 100 and let each session of a consumer constitute a minibatch (6,783 sessions/minibatches in total in the training set). The full model has 376,363 parameters to learn. For optimization of the loss function, we use the Adam algorithm with squared-root decay of learning rates.[12] To establish a benchmark, we also train and test the GRU using the centralized learning approach, that is, standard stochastic gradient descent on the full training set, where we use the same train/test split as in the FL setting, again with each session forming a minibatch. For computational efficiency, we choose $C = 0.2$; that is, 20% of randomly selected consumers work independently during each communication round. We also show the results obtained from setting $C = 0.1$ for comparison. We also vary the level of E, the number of local training epochs on each consumer's device using her local data

[12] For centralized learning, we set the learning rate η to $3 \times 1e-4$. For FL, η is set to 5 when the sampling rate $C = 0.2$ and the number of local training epoch $E = 1$; η is set to 3.3 when $C = 0.2$ and $E = 2$; η is set to 1.2 when $C = 0.1$ and $E = 1$. We trained over a wide range of learning rates, and these performed the best in terms of speed of convergence.

Table 3: Prediction Accuracy and Communication Rounds

Model	C	E	Recall@1	Communication rounds
Centralized Learning-GRU	-	-	0.60	-
Federated Learning-GRU	0.1	1	0.43	8,287
Federated Learning-GRU	**0.2**	**1**	**0.53**	**555**
Federated Learning-GRU	0.2	2	0.52	620

before communicating to the central server.

Table 3 reports the out-of-sample prediction accuracy as measured by Recall@1 averaged over all predicted clicks. We present the prediction accuracy levels for FL-GRU with various sampling rate C and local training epoch E. We also report the prediction accuracy obtained through the centralized learning approach as a baseline. When the sampling rate $C = 0.2$ and local training epoch $E = 1$, the FL achieves a prediction accuracy of 53%. That is, when we train the GRU using the FL approach, with 53% probability, the product with the highest predicted click probability is the actual product the consumer has clicked (out of 863 alternative products). The prediction accuracy obtained via the FL approach is comparable to that of the centralized approach, with the FL approach achieving 88% of the baseline prediction accuracy of the centralized approach.[13] We want to emphasize that with the FL approach, the central server/firm has never stored, accessed, or directly analyzed individual consumer data. Hence, the accuracy level of the FL is fairly impressive.

Computational costs are minimal in the FL setting because the size of the dataset stored in any single device is small while modern devices have fast processors. By contrast, communication costs are of major concern in distributed optimization settings such as the FL, because information needs to be passed back and forth between the nodes and the central server during the model optimization. In particular, limited upload bandwidth, network connection (3G, 4G, WiFi), and power plug-ins (battery) hinder unlimited communication. McMahan et al. (2017) show the following two elements of the FL may substantially reduce the number of communication rounds necessary for convergence:

[13] i.e., $0.53/0.60 = 0.88$.

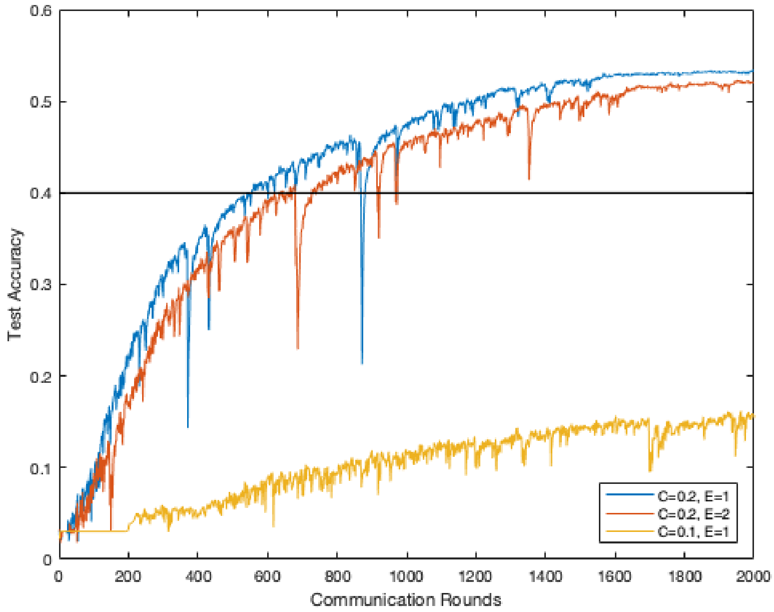

Figure 1: Test Accuracy for (i) $C = 0.1, E = 1$, (ii) $C = 0.2, E = 1$, and (iii) $C = 0.2, E = 2$. Plot for $C = 0.1, E = 1$ is only shown up to 2,000 communication rounds in order to compare the communication efficiency with the baseline of $C = 0.2$.

1. Increasing parallelism by increasing sampling rate: More consumers do computation independently during each communication round, and

2. Increasing computation at each consumer's node: Multiple updates are performed at the consumer level during each communication round.

We report in Table 3 the minimum number of communication rounds necessary to achieve a target accuracy of 40%. Figure 1 shows the learning curves, where the horizontal line represents the target 40% accuracy level. The target accuracy is reached after $8,287$ communication rounds when $C = 0.1, E = 1$. Increasing parallelism by setting C to 0.2 while maintaining $E = 1$ drastically reduces the number of rounds to only 555.[14] This figure also

[14] For the centralized learning, the GRU is trained on the full training set. The model parameters are updated iteratively and sequentially for each minibatch (i.e., simple stochastic gradient descent). For the centralized learning to achieve the 40% accuracy, 440 training epochs are necessary. One interesting analogy about communication is that if each minibatch update is counted as a communication round, the total number of communication rounds is $440 \times 6,873 = 3,024,120$. This number is much higher than the 555 rounds needed for the FL approach, implying a substantial computational burden.

includes results for $C = 0.2, E = 2$, which performs slightly worse than $E = 1$.[15]

5 Conclusion

Massive amounts of data generated by consumers provide a wealth of opportunities for firms to accurately predict consumer behavior and to target and provide customized services, thereby improving profitability as well as enhancing consumer experience. However, the rapid growth of the use of consumer data, along with recent data-breach incidents, has raised concerns regarding the protection of consumers' privacy. Governments in several countries are introducing regulations that greatly restrict firms' access, use, and sharing of consumer data. These regulations greatly restrict business activities of firms that rely heavily on consumer data for their business activities. Therefore, firms must find solutions to mitigate the impact of restrictive privacy regulations while keeping consumers' private data safe.

In this paper, we show how machine learning approaches allow firms to continue benefiting from vast amounts of consumer data without compromising consumers' privacy. Specifically, we discuss a recently developed FL approach, which uses a parallelized deep learning algorithm to train a model locally on each individual consumer's device. As an instantiation to demonstrate the applicability of this approach in a marketing setting, we build a session-based GRU recurrent neural network that predicts each consumer's click-stream under the FL framework. We show the prediction accuracy of the trained neural network via the FL approach is comparable to that of the benchmark centralized approach. Through this application, we demonstrate how firms can continue targeting consumers with a high level of accuracy without having to store, access, or analyze consumer data in centralized locations, thereby preserving consumers' sensitive information.

[15] As discussed in section 3, increasing local training epochs may not necessarily enhance communication efficiency. In McMahan et al. (2017), the authors draw the same conclusion.

References

Anderson, E. T. and D. Simester (2013): "Advertising in a Competitive Market: The Role of Product Standards, Customer Learning, and Switching Costs," *Journal of Marketing Research*, 50.

Barni, M., P. Failla, R. Lazzeretti, A.-R. Sadeghi, and T. Schneider (2011): "Privacy-Preserving ECG Classification With Branching Programs and Neural Networks," *IEEE Transactions on Information Forensics and Security*, 6, 452–468.

Bronnenberg, B. J., J. B. Kim, and C. F. Mela (2016): "Zooming In on Choice: How Do Consumers Search for Cameras Online?" *Marketing Science*, 35, 693–829.

Chen, Y. and S. Yao (2017): "Sequential Search with Refinement: Model and Application with Click-Stream Data," *Management Science*, 63, 4345–4365.

Cho, K., D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio (2014): "Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1724–1734.

Duchi, J. C., M. I. Jordan, and M. J. Wainwright (2012): "Privacy Aware Learning," *arXiv:1210.2085.*

Dwork, C., F. McSherry, K. Nissim, and A. Smith (2006): "Calibrating Noise to Sensitivity in Private Data Analysis," *In Theory of Cryptography Conference (TCC)*, 265-284.

Goldfarb, A. and C. E. Tucker (2011): "Privacy Regulation and Online Advertising," *Management Science*, 57, 57–71.

Hann, I.-H., K.-L. Hui, T. S. Lee, and I. Png (2003): "The Value of Online Information Privacy: An Empirical Investigation," Unpublished Manuscript.

Hidasi, B., A. Karatzoglou, L. Baltrunas, and D. Tikk (2016): "Session-Based Recommendations with Recurrent Neural Networks," *Published as a conference paper at ICLR.*

Hidasi, B. and D. Tikk (2016): "General Factorization Framework for Context-Aware Recommendations," *Data Mining and Knowledge Discovery*, 30, 342–371.

Hui, S. K., P. S. Fader, and E. T. Bradlow (2009): "Path Data in Marketing: An Integrative Framework and Prospectus for Model Building," *Marketing Science*, 28, 320–335.

Jin, G. Z. (2018): "Artificial Intelligence and Consumer Privacy," *NBER WORKING PAPER SERIES.*

Johnson, G. (2013): "The Impact of Privacy Policy on the Auction Market for Online Display Advertising," *Working Paper.*

McMahan, H., E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas (2017): "Communication-Efficient Learning on Deep Networks from Decentralized Data," *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 54.

Moe, W. W. and P. S. Fader (2004): "Dynamic Conversion Behavior at E-Commerce Sites," *Management Science*, 50, 326–335.

Mohassel, P. and Y. Zhang (2017): "SecureML: A System for Scalable Privacy-Preserving Maching Learning," Working Paper.

Montgomery, A. L., S. Li, K. Srinivasan, and J. C. Liechty (2004): "Modeling Online Browsing and Path Analysis Using Clickstream Data," *Marketing Science*, 23, 579–595.

Park, Y.-H. and P. S. Fader (2004): "Modeling Browsing Behavior at Multiple Websites," *Marketing Science*, 23, 280–303.

Rafieian, O. and H. Yoganarasimhan (2018): "Targeting and Privacy in Mobile Advertising," *Working Paper.*

Rubinstein, B. I. P., P. L. Bartlett, L. Huang, and N. Taft (2012): "Learning in a Large Function Space: Privacy-Preserving Mechanisms for SVM Learning," *J. Privacy and Confidentiality*, 4.

Sarwate, A. D. and K. Chaudhuri (2013): "Signal Processing and Machine Learning with Differential Privacy: Algorithms and Challenges for Continuous Data," *Signal Processing Magazine*, 30, 86–94.

Seiler, S. and S. Yao (2017): "The Impact of Advertising Along the Conversion Funnel," *Quantitative Marketing and Economics*, 15.

Shokri, R. and V. Shmatikov (2015): "Privacy-Preserving Deep Learning," *Proceedings of the 22nd ACM SIGSAC Conferences on Computer and Communications Security*, CCS' 15.

Sweeney, L. (2000): "Simple Demographics Often Identify People Uniquely," *Carnegie Mellon University Data Privacy Working Paper.*

Tang, J., A. Korolova, X. Bai, X. Wang, and X. Wang (2017): "Privacy Loss in Apple's Implementation of Dierential Privacy on MacOS 10.12," available at https://arxiv.org/abs/1709.02753.

Tucker, C. E. (2014): "Social Networks, Personalized Advertising, and Privacy Controls," *Journal of Marketing Research*, 51, 546–562.

Valentino-DeVries, J., N. Singer, M. H. Keller, and A. Krolik (2018): "Your Apps Know Where You Were Last Night, and They're Not Keeping It Secret," *New York Times*, Dec. 10.

Xie, P., M. Bilenko, T. Finley, R. Gilad-Bachrach, K. Lauter, and M. Naehrig (2014): "Crypto-Nets: Neural Networks over Encrypted Data," *arXiv:1412.6181*.

Yao, S. and C. F. Mela (2011): "A Dynamic Model of Sponsored Search Advertising," *Marketing Science*, 30, 447–468.

Yao, S., W. Wang, and Y. Chen (2017): "TV Channel Search and Commercial Breaks," *Journal of Marketing Research*, 54, 671–686.