




Customer Acquisition via Explainable Deep Reinforcement Learning

Yicheng Song,^{a,*} Wenbo Wang,^b Song Yao^c

^a Carlson School of Management, University of Minnesota, Minneapolis, Minnesota 55455; ^b Marketing Department, Hong Kong University of Science and Technology Business School, Clear Water Bay, Kowloon, Hong Kong; ^c Olin Business School, Washington University in St. Louis, St. Louis, Missouri 63130

*Corresponding author

Contact: ycsong@umn.edu,  <https://orcid.org/0000-0002-9107-814X> (YS); wenbowang@ust.hk,  <https://orcid.org/0000-0002-4288-0929> (WW); songyao@wustl.edu,  <https://orcid.org/0000-0002-0339-9649> (SY)

Received: September 20, 2022

Revised: March 30, 2023; February 21, 2024;
April 15, 2024

Accepted: April 20, 2024

Published Online in Articles in Advance:
May 21, 2024

<https://doi.org/10.1287/isre.2022.0529>

Copyright: © 2024 INFORMS

Abstract. Effective customer acquisition heavily hinges on sequential targeting to ensure that appropriate marketing messages reach customers. Sequential targeting could guide customers through the acquisition process and thus, optimize long-term revenue for the firm. Toward this goal, reinforcement learning (RL) has demonstrated great potential in facilitating sequential targeting during user acquisition. However, decisions made by RL during this process often lack explainability. We introduce the deep recurrent Q-network with attention model, which optimizes the long-term reward of sequential targeting while enhancing the explainability of the decisions. The key idea of the proposed model is to revise Q-learning by adding an attention mechanism to create a bottleneck, forcing the model to focus on features of the next ad exposure that will lead to optimal long-term rewards. We estimate our model using a comprehensive data set from a digital bank. The empirical results show that the proposed model is explainable and also outperforms state-of-the-art methods in terms of long-term revenue optimization. Specifically, the attention mechanism within the model functions as forward planning. The forward planning can spot those features in the next ad exposure that are more likely to lead to the optimal outcome. We further demonstrate how the model makes targeting decisions of advertising channel choices by showing that the model can (1) learn optimal ad channels to target customers from different industries, (2) adjust advertising channels in response to dynamic customer behaviors, and (3) learn the seasonality of the customer's industry and calibrate the ad channel correspondingly.

History: Olivia Liu Sheng, Senior Editor; Huimin Zhao, Associate Editor.

Funding: The authors thank General Research Fund [Grant 16500022] of Hong Kong Research Grants Council, [Grant RMGS20EG01-F] from HKUST, and [Grant WEB19EG01-F] for supporting this research.

Supplemental Material: The online appendix is available at <https://doi.org/10.1287/isre.2022.0529>.

Keywords: explainable reinforcement learning • customer acquisition • DRQN-attention • long-term revenue optimization • advertising channel choice

1. Introduction

Successful customer acquisition stands as a pivotal factor in the growth of businesses in the digital era. This process involves identifying potential customers and guiding them through the purchase funnel (Song et al. 2022). Companies gauge the effectiveness of their customer acquisition strategies by assessing the cost and efficiency of acquisition, considering them as key metrics (Chuck 2022). Notably, the customer acquisition costs on many digital platforms are exceptionally high, emphasizing the critical importance of ensuring that these costs do not exceed the average lifetime value of acquired customers (Hoffman and Novak 2000). The significance of customer acquisition costs persists, even in emerging industries that leverage cutting-edge technologies, such as artificial intelligence and cloud computing (AppsFlyer 2021).

The randomized experiment is a popular choice to evaluate the effects of various intervention policies for customer acquisition, including targeting different groups of customers (Frick et al. 2023) and using different ad formats and channels when deploying the ad (Reiley et al. 2012). However, these experiments often focus on examining the impact of each individual intervention, neglecting the potential interplay among them. This might lead to suboptimal results (Song and Sun 2023), especially for the settings that need to consider a sequence of interventions, such as sequential targeting. Furthermore, when the goal of an experiment is to find the optimal sequence of many interventions, the policy space grows exponentially, making it challenging to overcome the “curse of dimensionality” and identify the optimal policy using experiments. Fortunately, as suggested by the causal decision-making literature

(Fernández-Loría and Provost 2022), an accurate causal effect estimation of each intervention via randomized experiments is often unnecessary for determining which interventions to use and their sequence so as to optimize the outcomes. Instead, the optimization only requires differentiating the effectiveness of different interventions under different contexts. Therefore, as an alternative to randomized experiments, a model-based approach aiming to achieve desired business outcomes could be a promising solution to the customer acquisition problem.

Machine learning models have become essential in developing cost-effective customer acquisition strategies. Current machine learning-based targeting techniques utilize supervised learning models to predict the profitability of prospects under different individual marketing interventions (Simester et al. 2020). However, the customer's decision to purchase a product or service often entails a significant commitment of time and money, making the conversion less likely to be triggered by a single intervention. Recognizing this, prior work (Schwartz et al. 2017) frames customer acquisition as converting prospects through a series of marketing messages sequentially across diverse contexts (e.g., different devices and websites). Allocating these sequential marketing messages presents a challenge as the responses of prospective customers to different messages are either unknown or only partially revealed. Understanding these responses improves over interactions, leading to a "learn-and-earn" or "exploration-and-exploitation" trade-off. Firms must strategically deploy various marketing messages, not only leveraging known responses for stable revenue but also exploring new messages with uncertain yet potentially more profitable outcomes. To address this objective, researchers have turned to reinforcement learning (RL), a framework designed to allocate a limited budget in a way that balances exploration and exploitation.

RL models have shown promise in addressing various business challenges, such as optimizing the ad click-through rate (CTR) (Hauser et al. 2009), designing sequential promotions (Wang et al. 2023), and guiding subject allocations for future trials based on past randomized experiments (Song and Sun 2023). These applications of RL demonstrate its versatility and effectiveness in making optimal decisions in complex business environments. Despite the successes, most RL models developed to facilitate business decision making are black-box machine learning models without transparency, which leaves users of such models with a limited understanding of why particular decisions are made. However, it is crucial for RL models adopted by firms to be transparent and easily understandable pertaining to how decisions are made, what information is used, and why mistakes occur (Puiutta and Veith 2020). Otherwise, the lack of explainability can frustrate and

confuse users and diminish their trust in intelligent systems, hindering the future applications of these models (Zhang and Curley 2018). Consequently, there has been a growing demand for explainable models. For example, DARPA started the Explainable Artificial Intelligence program in 2018 to create explainable high-performing models and to convey the reasoning process of machines that humans may understand (Turek 2018). Recently, information systems researchers also called for empowering machine learning models with explainability to address the lack of transparency in deep learning (Berente et al. 2021). Note that such explainability does not necessarily need to be clean causal effects identified from field experiments. Instead, it can be insights that enable researchers to glimpse into the decision-making process via the outputs of the machine learning models.

In this study, we present the deep recurrent Q-network with attention (DRQN-attention) model, designed to enhance the explainability of RL models for customer acquisition while preserving the primary objective of optimizing long-term reward.¹ The model aims to expose the right information to the right prospect in the right context. It is the prospects themselves who decide whether to apply for the service or buy the product. The model adopts the deep recurrent Q-network (DRQN) (Hausknecht and Stone 2015) to model the partially observable Markov decision process (POMDP) of prospects, where customer states are only partially observed and cannot be fully represented by a few recent observations. To create a bottleneck in the RL agent and compel the model to focus on task-relevant information, the proposed model integrates the Q-learning algorithm with attention mechanism (Vaswani et al. 2017). The customized attention mechanism allows direct observation of the information used by the model for decision making, rendering the model more explainable than conventional RL models.

We review the related literature in Section 2 to identify the research gap and summarize the contributions of this study. Following that, we introduce the empirical context of this study in Section 3. In Section 4, we develop the DRQN-attention model that integrates a customized attention mechanism within DRQN. The evaluation of the DRQN-attention model on a large data set collected from a digital bank is presented in Section 5, where we demonstrate the model's superiority over state-of-the-art methods for long-term revenue optimization. To address privacy concerns, we adopt federated learning (FL) for the RL models in Section 5.3 and show-case that the model can be applied in privacy-sensitive settings without compromising service quality. In Section 6, we delve into gaining more insights into the attention mechanism results, demonstrating that the attention mechanism functions as forward planning. This forward planning identifies which features of the

next ad exposure opportunity are more likely to lead to optimal outcomes. Further insights into the attention mechanism are explored in Section 7, where we illustrate that the attention mechanism helps managers better understand how the model chooses online advertising channels. This includes optimizing advertising channels for prospects from different industries, adjusting advertising channels to dynamic prospect behaviors, and developing sequential advertising channel plans that account for the seasonality of prospects from the agricultural industry.

2. Literature

Our research closely relates to two streams of literature. The first is the studies that develop RL models to facilitate business decision making and problem-solving. The second stream is explainable RL, which is methodologically related to this study.

2.1. Reinforcement Learning for Business Decision Making

RL has the potential to be a powerful tool for optimizing business goals by allocating limited resources effectively. In marketing, RL can be used to learn dynamic and adaptive marketing strategies by continuously learning from past experiences and fine-tuning the marketing messages to achieve optimal outcomes. With its ability to balance exploitation and exploration of marketing messages, RL can help companies drive better outcomes. Multiarmed bandit (MAB) is the first RL model adopted by researchers that optimizes the allocation of limited resources to facilitate marketing strategies (Hauser et al. 2009). Although MAB can help firms optimize outcomes, such as CTR, a major limitation of such models is that they mainly focus on the immediate feedback from the prospects (e.g., ad click) but do not explicitly model long-term revenues.² However, ad click is just an intermediary step toward customer acquisition, and firms are more interested in optimizing long-term revenues that consider both intermediary steps and the final goal (e.g., customer acquisition). Recognizing the importance of optimizing long-term revenues, new RL models have been developed. Building on the Q-learning framework (Watkins and Dayan 1992), Wang et al. (2023) propose a deep RL model for sequential targeting problems. This model first builds a predictive model to capture customers' responses to various marketing actions and then trains a deep RL agent to interact with the predictive model. The goal is to learn optimal sequential marketing strategies, considering the dynamic sequential behavior of customers to optimize long-term revenues. In a similar vein, Song and Sun (2023) develop an RL algorithm within the Bayesian framework. This algorithm learns the returns to sequential intervention strategies from historical randomized

experiments, guiding subject allocation in future experiments to further improve intervention strategies.

From the perspective of model explainability, current approaches in this field mostly attempt to explain model decisions through post hoc analysis. Specifically, they normally create a black-box RL model and then utilize post hoc analysis to demonstrate its explainability, showing that the variation of model outputs can be traced back to changes in model inputs. However, little advance has been made for intrinsic explainable RL models, especially for the settings of customer acquisition. Unlike post hoc explanations, the intrinsic explainability of RL is achieved by designing models that are self-explanatory and incorporate explainability directly into their structures.³ These explainable models are either globally explainable or can provide explanations for individual prediction, thereby offering accurate and undistorted explanations of model decisions (Du et al. 2019). In sum, we propose an intrinsic explainable RL model that provides explainability for customer acquisition, addressing the gap in the literature and meeting the needs of businesses.

2.2. Explainable Reinforcement Learning

RL has been successfully applied to a variety of real-world scenarios, including autonomous vehicles (Kiran et al. 2022), career planning (Kokkodis and Ipeirotis 2021), and other domains that require sequential decision making. However, making RL models explainable is challenging because of two main factors. First, RL models are often built using black-box machine learning models, inherently lacking explainability. Second, the design of RL models for sequential decision making and long-term revenue optimization inevitably makes them difficult to explain.

Attention mechanisms have gained popularity in deep learning for several reasons (Brauwers and Frasin-car 2021). Most attention mechanisms can be trained jointly with a base neural network model using regular back propagation. Additionally, attention introduces a form of explanation into neural network models, which are generally known for being highly complex and challenging to explain. Recent advancements in state-of-the-art explainable RL models have, therefore, incorporated attention mechanisms. As per the definition of the general attention model in Brauwers and Frasin-car (2021), there are three major components in attention mechanisms: query q , key K , and value V . The query, designed based on the desired output of the model, instructs the attention model on which features to focus. It serves as a request for information or a question. Two separate matrices, keys matrix K and values matrix V , are then generated from the input features. The attention module's goal is to produce a weighted average of the value vectors in V weighted by the attention weight a . The attention weight a results from the interaction between

the query q and the keys matrix K through different attention score functions. The hypothesis behind modeling attention is that the magnitude of attention weight a correlates with the relevance of a specific input region to the model prediction. Visualizing attention weight a for a set of input and output pairs provides insights into the importance of different inputs for model decisions. The general attention model is versatile and can be applied to various problems. However, researchers need to tailor the attention mechanism (i.e., query q , key K , and value V) to address specific research questions in different contexts (Chaudhari et al. 2021). Next, we will introduce representative RL models that incorporate attention mechanisms and discuss the differences between their attention models and the one proposed in this study.

Many explainable RL models leveraging attention mechanisms were developed in the context of video games, where the zero-cost game stimulator ensures efficient RL model training, and well-structured game rules facilitate feasible and widely accepted model decision interpretation (Torrado et al. 2018). Specifically, Annasamy and Sycara (2019) integrate an attention mechanism with the deep Q-network, which provides explanations for how an intelligent agent makes decisions when playing video games. To provide an understanding of the learned gameplay policy, the authors visualize clusters of state to represent attention maps for any action-value pair. Related, Mott et al. (2019) introduce a spatial attention mechanism to visualize information in an actor-critic RL model. By exploiting the attention maps on the spatial basis, one can understand how the RL model solves a task in the video game, identifying the type of entities that the agent attends (“what”) and spatial locations where the agent directs its attention (“where”). Shi et al. (2020) adopt a self-supervised explainable network to produce fine-grained attention masks to highlight task-relevant information in the game that most likely leads to the agent’s decisions. Itaya et al. (2021) integrate an attention mechanism into the asynchronous advantage actor-critic model to analyze the decision making of an intelligent agent. The authors introduce attention mechanisms into both the policy and value branches of the actor-critic model, using attention to reveal the reasons behind the two branches.

In line with these studies, we have developed an explainable RL model for customer acquisition by incorporating a customized attention mechanism. However, different from these models developed in the context of video game control, the proposed model takes into account the nuances of customer modeling, rendering the customized attention mechanism different from previous works in several key dimensions.

1. For studies using video game play as the context, the latest video image frame generally contains complete

information about the environment (e.g., Go Game, Atari game). Therefore, most models utilize the latest video image frame to construct query, key, and value in the attention mechanism. However, in the context of customer acquisition, the potential customer’s state can rarely be described only by the latest observation. Accordingly, we use recurrent neural network (RNN) to process potential customers’ historical interactions, along with the static features, to construct the query.

2. Because existing studies use video game image frames to construct query, key, and value in the attention mechanism, they apply the attention weight to the feature map generated by a convolutional neural network (CNN), and then, they use a deconvolution function to restore the attention weight back to the original image space. However, in the context of customer acquisition, there is no reverse engineering function when using RNN to process historical interactions with the customer. Hence, we apply the attention map directly to the raw incoming interaction to highlight important features.

3. Different from just utilizing video game frames (i.e., same source), we rely on different sources to construct the query, key, and value in the attention mechanism. The historical interactions and static information underpin the query, and the incoming interaction generates key and value. The rationale is that given static information and that historical interactions of a prospect are fixed and immutable, the model’s focus is directed toward recognizing incoming interactions for long-turn reward optimization. This step allows the model to dynamically adjust the attention weight based on different historical interactions and static information, discovering features in the incoming interaction that lead to more promising outcomes.

Although many works on explainable RL with attention mechanisms are developed in the context of video games to facilitate model training and decision interpretation, there are related works that have explored other domains. For instance, Zhang et al. (2021) propose an interpretable deep reinforcement learning model for diagnosing Alzheimer’s disease. They apply the attention mechanism to magnetic resonance imaging images and clinical information, thereby explaining parameter weights for Alzheimer’s disease diagnosis. Fei et al. (2021) develop an explainable RL model to automatically optimize attention distribution on sequential textual data to minimize natural language processing (NLP) task training losses. Their results demonstrate that the model can yield more reasonable attention distribution for NLP tasks. Additionally, Wang et al. (2019) focus on the quantitative stock trading problem, aiming to enhance investment strategy through an interpretable RL model. They customize the attention mechanism to capture the interrelationship among stocks, enriching

investment strategy with additional information. These context-specific studies demonstrate how attention mechanisms can be customized to fit specific research questions. Table 1 summarizes the detailed differences in attention mechanisms across these studies.

To the best of our knowledge, there is no existing intrinsic explainable RL model specifically designed for customer acquisition problems. Our proposed model addresses two critical challenges. (1) It provides a general RL model for customer acquisition that can help companies optimize their long-term revenues. Additionally, (2) it enables managers to gain valuable insights into the decision-making processes of the model. The intrinsic explainability of our proposed model sets it apart from other existing models, allowing managers to understand informed decisions based on the results generated by the model.

3. Research Context

We partnered with a digital bank in China for this research. Our partner provides online financial services to underbanked and unbanked small- and medium-sized enterprises (SMEs) in China, a market that is highly overlooked by traditional banks. As highlighted in the World Bank report (Bank 2019), SMEs play a vital role in most economies, particularly in developing countries. They constitute a significant portion of businesses globally and make substantial contributions to job creation and global economic growth. Nevertheless, access to financial services remains a significant constraint for SMEs and ranks as the second most frequently cited obstacle hindering SMEs' growth. The International Finance Corporation estimates that a staggering 65 million firms, equivalent to 40% of SMEs in developing nations, face an unmet financing gap of \$5.2 trillion annually, which is 1.4 times greater than the current global lending level to SMEs (Khanna et al. 2017). Therefore, the development of an efficient customer acquisition model that accurately targets potential

SMEs clients and delivers essential financial information becomes imperative.

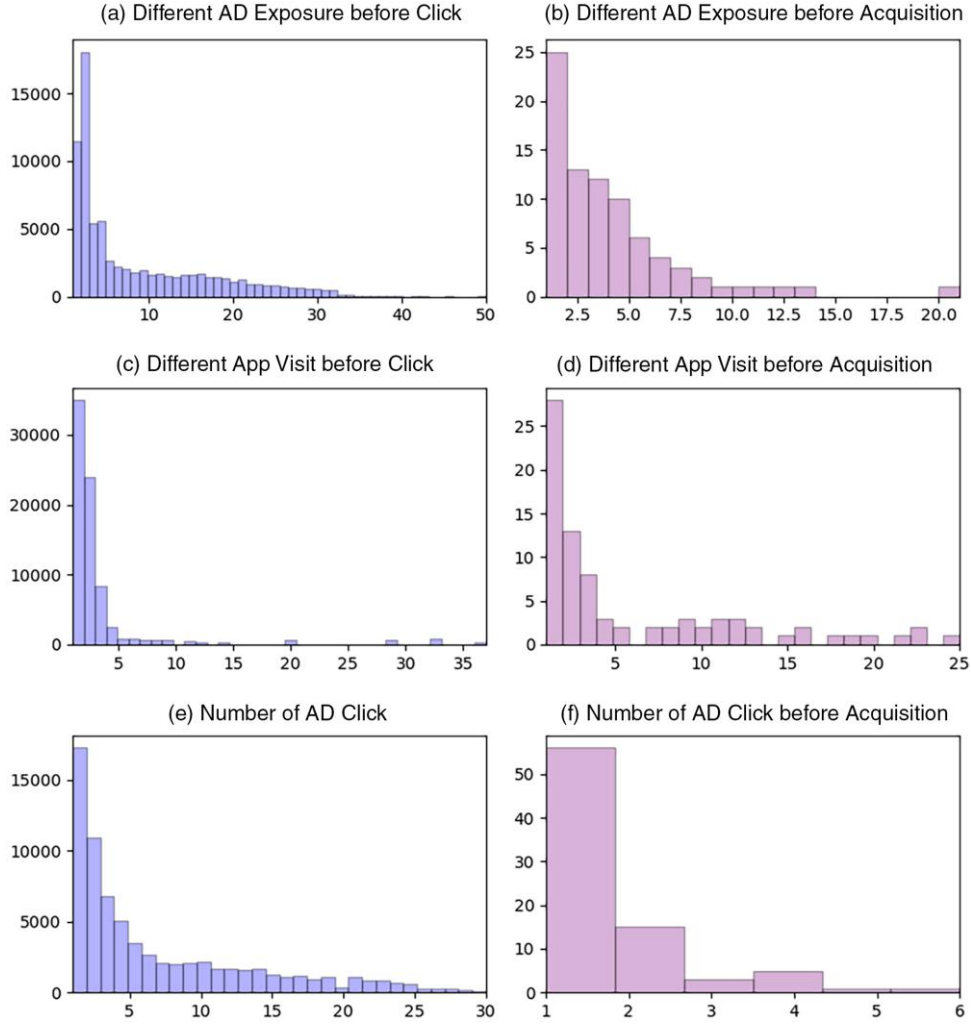
Our partner's advertising channels encompass numerous popular third-party mobile apps, with the customer acquisition team aiming to identify prospective borrowers by delivering various advertising messages across these channels. Their ad library comprises nearly 200 different ad messages, each highlighting different aspects of their product, such as speedy loan approval within 15 minutes, daily interest rates as low as 0.01%, and loans specialized for small and microbusinesses, among others. It is essential to underscore that their loan service for SMEs is a standard public service, treating all individuals equally without differential treatment, such as offering higher interest rates based on personal attributes. Additionally, the service is accessible to anyone, with prospects making their own decisions about whether to apply for the service. Their existing customer acquisition system is developed based on contextual MAB (Lu et al. 2010) to optimize the CTR. When a potential customer visits one of the advertising channels, an ad exposure request is sent to the bank. The customer acquisition system then responds with decisions on whether to show an ad to the prospect by evaluating the cost and expected gain and if affirmative, which ad to display. If an ad is shown, the prospect can click on it to obtain more information about the service and potentially apply for credit, marking a successful customer acquisition.

We collected clickstream data from a random sample of the bank's potential customers. Examining the activities of these potential customers reveals substantial heterogeneity in their responses to ads and paths to customer acquisition (i.e., applying for credit). Figure 1, (a) and (b) suggests that potential customers often receive multiple different ads before they click and apply for credit, implying that the combinations of different ads help move the prospects through the customer acquisition funnel. In addition, potential customers are active in multiple advertising channels as they have been receiving ads on different apps (Figure 1,

Table 1. Attention Mechanism Components (Query, Key, and Value) Comparison

	Setting	Query	Key	Value
Annasamy and Sycara (2019)	Video game	CNN on video frames	Action specific query, randomly generated	Action specific value, randomly generated
Mott et al. (2019)	Video game	ConvLSTM on game frames	CNN on game frames + spatial basis	CNN on game frames + spatial basis
Shi et al. (2020)	Video game	Attention weight from autoencoder model	Attention weight from autoencoder model	Latest raw game frames
Itaya et al. (2021)	Video game	CNN on video frames	CNN on video frames	CNN on video frames
Zhang et al. (2021)	Medical	MRI	MRI	MRI
Fei et al. (2021)	NLP	Word embedding	Word embedding	Word embedding
Wang et al. (2019)	Stock trading	Stock representation	Stock representation	Stock representation
This study	Customer targeting	Historical interaction + static info	NN on incoming interaction	Raw incoming interaction

Note. ConvLSTM, convolutional long short-term memory network; MRI, magnetic resonance imaging; NN, neural network.

Figure 1. (Color online) Prospect Behaviors Before Ad Click and Customer Acquisition

Notes. Panels (a) and (b) suggest that prospects often receive multiple different ads before clicking and applying for credit (customer acquisition). Panels (c) and (d) indicate that prospects are active in multiple advertising channels. Although there are many ad clicks in panel (e), only a few of them lead to customer acquisitions in panel (f), and many acquisitions are preceded by multiple ad clicks.

(c) and (d)), indicating the necessity to target them across channels. Finally, even though we observe many ad clicks (Figure 1(e)), only a handful of the clicks lead to customer acquisitions (Figure 1(f)). Moreover, as 32% ($\approx 26/81$) of acquisitions are preceded with more than one click, indicating that CTR alone may not be a good metric, the targeting should be optimized by comprehensively considering both the final outcome (acquisition) and the intermediate steps (ad click). These observations reinforce our motivation to develop an intelligent model that meets the needs of crosschannel prospect targeting to optimize long-term revenue.

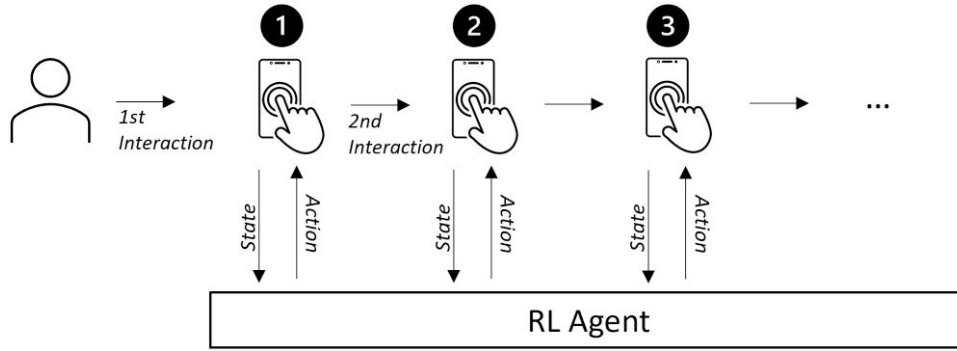
4. Model

We frame customer acquisition as an RL problem, where an intelligent RL agent interacts with potential

customers by deploying a sequence of ad messages over time. As illustrated in Figure 2, whenever a potential customer interacts with a collaborated channel,⁴ an ad exposure request that summarizes the state of the prospect is sent to the RL agent, and the agent will return with a decision of whether to show an ad and if yes, which ad to show. The RL agent's objective is to optimize long-term rewards from prospects. Next, we formalize the three key elements (S, A, R) of the RL model in this setting as follows.

- State S . A state $s_{it} \in S$ summarizes the state of prospect i at the t th interaction, which consists of her interaction history (i.e., interactions $1, \dots, (t-1)$) and the contextual information of the current t th interaction (e.g., website, app, time ...). Once we observe the prospect's subsequent interaction (i.e., $t+1$), the prospect's state will transit to $s_{i(t+1)}$.

Figure 2. Customer Acquisition via RL-Enabled Sequential Targeting



Notes. State indicates the ad exposure request that summarizes the state of the customer at the current interaction. Action indicates whether to show an ad and if yes, which ad to show.

- **Action A .** The action space A is a collection of ads as well as the option of not showing an ad (“no ad”). Accordingly, action in the t th interaction with prospect i , $a_{it} \in A$, is showing a specific ad or “no ad.”
- **Reward R .** If the action a_{it} shows an ad, given the prospect’s state s_{it} , the intelligence agent will receive an immediate reward $r_{it} \in R$ based on the prospect’s feedback. The prospect might click the ad and further register as a customer (customer acquisition), which leads to positive rewards.⁵ Otherwise, a negative reward is incurred because of the cost of displaying the ad. If the action is “no ad,” r_{it} will be zero.

With the outlined components, we can formally define customer acquisition within the RL framework. When observing the state s_{it} of prospect i , RL aims to learn a value function $\mathbb{F}(s_{it}, a_{it})$ that estimates the expected cumulative reward by applying action a_{it} under the state s_{it} . This estimation goes beyond the immediate reward r_{it} and encompasses rewards in the future. Consequently, the RL model is inclined to select action a_{it} that leads to the optimal value of $\mathbb{F}(s_{it}, a_{it})$, indicating a promising outcome. Next, we will explore various options of value function $\mathbb{F}(s_{it}, a_{it})$. It is noteworthy that the RL framework described provides a general solution to customer acquisition problems, irrespective of specific contexts.

4.1. Deep Recurrent Q-Network

Q-learning is a classical RL algorithm that estimates the cumulative long-term expected reward of executing an action for a given state (Watkins and Dayan 1992). Such estimated long-term rewards are known as Q values. A higher Q value $Q(s_{it}, a_{it})$ indicates that an action a_{it} is deemed to yield a better long-term reward given the state s_{it} . For an episode containing an ordered trajectory of state, action, and reward $\{s_{it}, a_{it}, r_{it}, s_{i(t+1)}\}$ obtained by executing the action a_{it} at the t th interaction, Q value $Q(s_{it}, a_{it})$ is defined as the observed immediate reward r_{it} plus the max Q value across all actions in the next

state $s_{i(t+1)}$:

$$Q(s_{it}, a_{it}) = r_{it} + \gamma \max_{a_{i(t+1)}} Q(s_{i(t+1)}, a_{i(t+1)}), \quad (1)$$

where $\gamma \in [0, 1]$ is the discount factor that balances current and future rewards. Thus, the goal of the RL agent is to learn policy $S \rightarrow A$, which chooses the action a_t that leads to the optimal Q value in each state. In many real-world settings, the space that describes the state and action is often too large to estimate Q values for all $|S| \times |A|$ pairs. Accordingly, as a general and flexible model, the deep neural network can be used to learn the complex state representation. Therefore, combining Q-learning with a deep neural network (i.e., deep Q-network (DQN)) can be used to accommodate complex inputs in real-world settings and learn actions to optimize the long-term reward. Such a model has been adopted by researchers to optimize sequential promotion design (Wang et al. 2023). Specifically, they model DQN as a neural network with parameters θ , and Q values can be estimated using the function $Q(s_{it}, a_{it} | \theta)$. To train such a neural network, the parameters θ are chosen to minimize the temporal difference via the following loss function:

$$\begin{aligned} \mathcal{L}(S, A, R | \theta) &= \sum_{i,t} \left(Q(s_{it}, a_{it} | \theta) - \left(r_{it} + \gamma \max_{a_{i(t+1)}} Q(s_{i(t+1)}, a_{i(t+1)} | \theta) \right) \right)^2. \end{aligned} \quad (2)$$

One underlying assumption of DQN is the Markov decision process (MDP), where the future state is based solely on the most recent action and state: $P(s_{i(t+1)} | s_{it}, a_{it}, \dots, s_{i1}, a_{i1}) = P(s_{i(t+1)} | s_{it}, a_{it})$. Such a Markov property, however, rarely holds in the real world, especially because a prospect’s state can rarely be fully described by only the most recent observations. As an alternative to MDP, the POMDP better describes many real-world environments by acknowledging that the observation received by the agent is only a partial glimpse of the state. To demonstrate the advantage and feasibility of

POMDP, Hausknecht and Stone (2015) propose DRQN, a combination of RNN and deep Q-learning. Different from DQN, which only uses the most recent video game frames to summarize the state of the environment, DRQN utilizes RNN to process sequential video game frames so as to summarize the state more comprehensively (Hausknecht and Stone 2015). Similarly, the state of a prospect is also a POMDP as it cannot be summarized by only the most recent observations. Thus, Song and Sun (2023) also utilize DRQN to process the interaction history of the prospect to summarize the state. Following these prior studies, we can adopt DRQN to learn the optimal customer targeting policies.

Figure 3 outlines the framework for applying DRQN to our setting. RNN is used to process historical interactions, where hi_{it} describes the agent's t th interaction with prospect i that consists of the contextual features of the interaction (including time, device, type of location (TOL), app, prospect reaction (click or no click), etc.).⁶ With interaction sequence $\{hi_{i1}, \dots, hi_{it}\}$ associated with prospect i , a gated recurrent unit (GRU) is utilized to process the interaction sequence to learn the state s_{it} :

$$s_{it} = \text{GRU}(hi_{i1}, hi_{i2}, \dots, hi_{it}). \quad (3)$$

The detailed mechanism of GRU is specified in Online Appendix C. The state s_{it} is used to predict the Q values $Q(s_{it}, a_t | \theta_{RNN})$ of executing different action a_t given the state. The parameter θ_{RNN} of the DRQN model can be learned by minimizing the same loss function outlined in Equation (2). We also incorporate double Q-learning (Van Hasselt et al. 2016) and dueling network architecture (Wang et al. 2016) into the DRQN model. The main benefit of these variants is that they improve the

learning stability and generalize learning across actions without imposing any changes to the underlying RL algorithm.

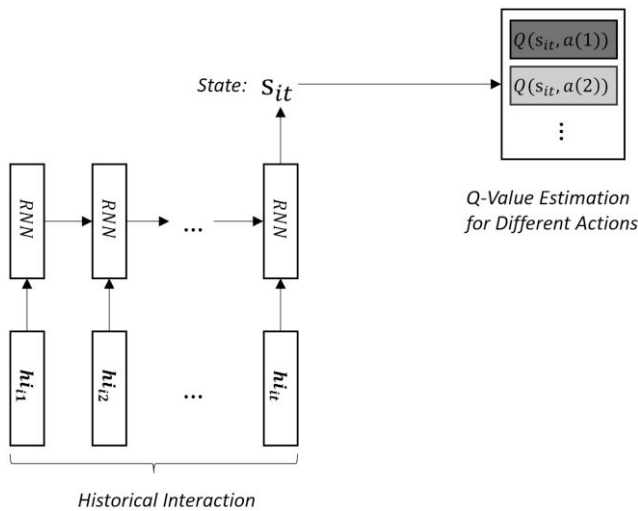
4.2. DRQN-Attention

RL has great potential for improving the long-term revenue of customer acquisition decisions. However, such complex models usually do not explain their decisions, which is a barrier to adoption. Therefore, researchers revise the RL models by adding explanation components to build intrinsic explainable models. Such intrinsic explainable models aim to achieve explainability while maintaining highly competitive performance (Wang et al. 2022). For instance, although the classical MAB uses a pure mechanical algorithm to select the next instance to trial, Cao and Leng (2021) integrate contextual bandit and decision tree into a unified model that adaptively collects new data points and provides explanations of model decisions by splitting user feature space via feature importance signaling.

Following this stream of research, we aim to empower the aforementioned RL models by explaining their decisions in a more intuitive fashion. Recently, the attention mechanism has been adopted to enhance the inherent explainability of machine learning models (Vaswani et al. 2017). Specifically, the attention mechanism is a technique that mimics the cognitive attention of human beings, which identifies “relevant” information from the input data and fades out the rest depending on the current cognitive task. As the attention layer explicitly weights input features, those weights can be used to identify which pieces of input features the model utilizes for different decisions. For example, Leng et al. (2020) adopt an attention mechanism within the geometric deep learning model to help increase the explainability of recommender systems on the social network. The attention mechanism has also been applied in the video game setting to enhance the explainability of the RL model's decisions during game plays (Mott et al. 2019).

Motivated by recent advancements in the literature, we endeavor to enhance the DRQN by integrating a tailored attention mechanism. Drawing on the general attention mechanism outlined in Section 2.2, our goal is to customize it based on the contextual characteristics of the customer acquisition problem. Specifically, we aim for the attention mechanism to dynamically adjust based on the prospect's static information and historical interactions. The objective is to identify task-relevant features from each incoming ad exposure opportunity, ultimately enhancing the customer acquisition policy. Given that static information and historical interactions are fixed and immutable, the model's focus is directed toward recognizing incoming interactions for targeting opportunities. In the upcoming section, we will first enumerate the various data relevant to this context.

Figure 3. Framework of Deep Recurrent Q-Network



Notes. Historical interaction data $\{hi_{i1}, hi_{i2}, \dots, hi_{it}\}$ associated with prospect i until the t th interaction will be processed by an RNN to get the prospect state representation s_{it} . We then predict the Q values of excusing different actions based on state s_{it} .

Subsequently, we will delve into how to leverage these data for the customized attention mechanism.

For the advertising opportunity of the incoming t th interaction with prospect i , DRQN-attention collects three parts of the data. (1) Static characteristics of the focal prospect $static_i$ describes the static characteristics of the focal prospect. (2) There are also historical interactions up to the $(t - 1)$ th interaction with prospect I (i.e., $\{hi_{i1}, hi_{i2}, \dots, hi_{i(t-1)}\}$). (3) f_{it} contains the features of the t th interaction with prospect i , which consists of contextual features describing the incoming ad exposure opportunity (e.g., app, location, connection type, time ...). f_{it} is represented via one-hot encoding as it is the target variable we will apply attention weight to, and the one-hot encoding schema of f_{it} makes attention weight explainable. The attention mechanism relies on the first two parts of the data (i.e., static characteristics and historical interactions) to place weights on the one-hot encoding representation of the features in the incoming interaction to learn the optimal targeting policy. Thus, with different static information and historical interactions, the model can automatically adjust the attention weights, signaling which features of the incoming interaction are used to make the decision.

As outlined in Figure 4, we customize a (key, value, and query) system for the attention mechanism to weigh the features of the incoming interaction based on

static characteristics and historical interactions. All $|J|$ features of the incoming interaction $\{f_{it}(1), \dots, f_{it}(|J|)\}$ are candidates to which we will place the attention weight. We first encode these features into a set of key-value pairs. We apply a dense layer $\mathbb{K}(x)$ on f_{it} to get a $k \times |J|$ key value matrix K_{it} , where the j th scalar feature in f_{it} will be transformed to a k -dimensional key column $K_{it}(j)$. The value vector v_{it} will be set as the copy of f_{it} . Then, we generate the query vector qr_{it} via a query-generating dense function $\mathbb{Q}\mathbb{R}(x)$ by processing the concatenation of the prospect's static characteristics $static_i$ and historical embedding he_{it} (generated by RNNs that process the historical interactions $\{hi_{i1}, hi_{i2}, \dots, hi_{i(t-1)}\}$). qr_{it} is a k -dimensional vector that will interact with each column in the key matrix K_{it} . Therefore, the attention logit map $\widetilde{att}_{it}(j)$ in Equation (4) is the inner product between query vector qr_{it} and the j th column in $K_{it}(j)$:

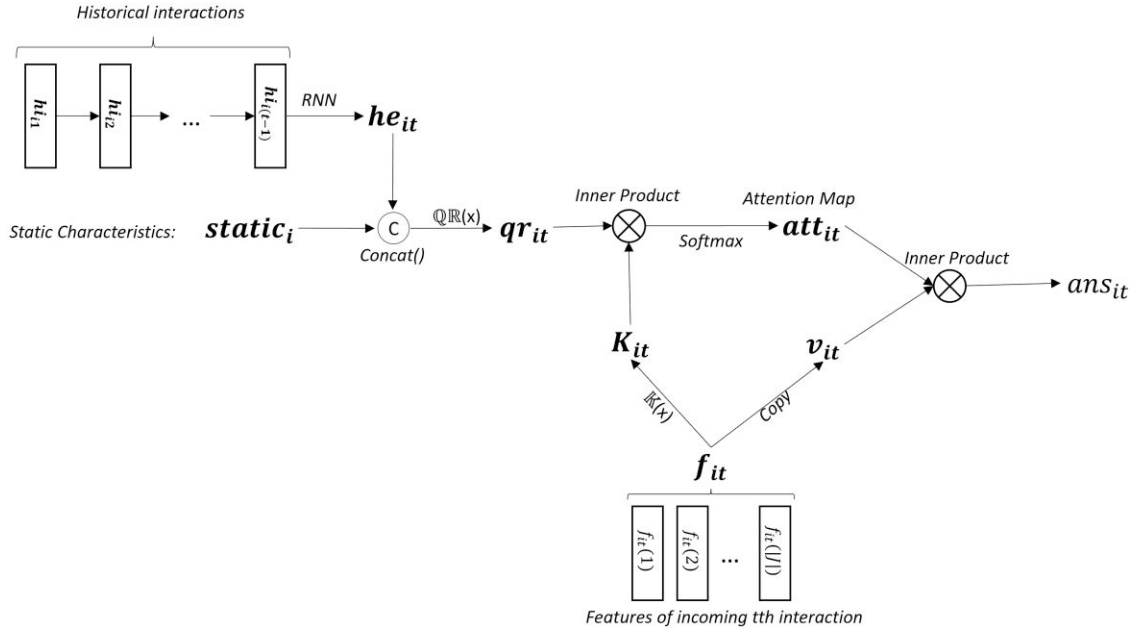
$$\widetilde{att}_{it}(j) = \frac{qr_{it}K_{it}(j)}{\sqrt{k}}. \quad (4)$$

We then take the softmax function on \widetilde{att}_{it} and get the normalized attention weight $att_{it}(j)$:

$$att_{it}(j) = \frac{\exp(\widetilde{att}_{it}(j))}{\sum_j \exp(\widetilde{att}_{it}(j))}. \quad (5)$$

Each attention weight $att_{it}(j)$ measures the importance of the j th feature of the incoming t th interaction when

Figure 4. Outline of the Customized Attention Mechanism in the DRQN-Attention Model



Notes. The customized attention mechanism uses static characteristics of the prospect and her historical interactions to query the features of the incoming interaction to identify task-relevant features. We first process historical interactions $\{hi_{i1}, \dots, hi_{i(t-1)}\}$ with an RNN to get historical embedding he_{it} . Then, we construct the query vector qr_{it} based on the concatenation of he_{it} with the static feature of the prospect $static_i$. Afterward, we generate the key value matrix K_{it} based on f_{it} , the one-hot encoding feature vector of the incoming interaction. We then apply the query vector qr_{it} to query each column in key matrix K_{it} . The result will enter a softmax function to generate the attention weight att_{it} across all $|J|$ features of the incoming interaction. Finally, we associate attention weight with value vector v_{it} (copy of f_{it}) to generate the answer ans_{it} of the query. The answer will be used to learn the state representation s_{it} of the prospect.

the RL model makes the targeting decision. Consequently, we create the answer ans_{it} to the query by multiplying $att_{it}(j)$ to the corresponding value $v_{it}(j)$ in v_{it} and summing over all $|J|$ features:

$$ans_{it} = \sum_{j=1}^{|J|} att_{it}(j)v_{it}(j). \quad (6)$$

The process describes one attention procedure. Following Vaswani et al. (2017), we adopt a multihead attention schema with n heads to generate n answers. Finally, all n answers $\{ans_{it}(1), ans_{it}(2), \dots, ans_{it}(n)\}$ and n queries $\{qr_{it}(1), qr_{it}(2), \dots, qr_{it}(n)\}$ will form as the input to state generating layer $\mathbb{S}(x)$ to learn the state representation s_{it} for prospect i :

$$s_{it} = \mathbb{S}(ans_{it}(1), \dots, ans_{it}(n), qr_{it}(1), \dots, qr_{it}(n)). \quad (7)$$

Similar to DRQN, the state s_{it} will be used to predict the Q value $Q(s_{it}, a_{it} | \Theta)$ for executing action a_{it} , where Θ is a set that contains parameters of RNN and customized attention mechanism. Thus, we can seamlessly integrate the attention mechanism into the Q -learning framework to learn the optimal targeting policy via Equation (1). Several properties of the proposed DRQN-attention model are worth highlighting. First, the model is fully differentiable. All parameters in Θ can be trained via back propagation by minimizing the loss function in Equation (2). Second, the query vectors are a function of static characteristics and historical interactions of the focal prospect; this allows an automatic “forward-planning” mechanism where the query function can actively query all the features in the next ad exposure opportunity, aiming to optimize the long-term reward. This process will unveil important features in the next ad exposure for decision making. In Section 7, we will show how attention weights can guide the model to choose advertising channels under different scenarios.

5. Empirical Analysis

5.1. Data

A random sample of 50,000 prospects is selected from the potential customer pool of the bank. For each prospect, the bank collects her or his static characteristics, including age, gender, residential city, and marital status. The clickstream data associated with these prospects are also collected over the period of June to December 2019. Visits to all advertising channels (third-party mobile apps) of each prospect are tracked via device identification. Upon a prospect’s visit to any advertising channels, the contextual features of this interaction are sent to the bank’s server. These contextual features include the date and time, time gap (in minutes) to the previous interaction, app identification, internet connection type, device type, and Type of Location.⁷ After receiving the contextual features, the company’s customer acquisition system (contextual

multiarmed bandit) decides whether to show an ad and if yes, which ad to show. If an ad is displayed, the prospect may ignore the ad or click the ad. After clicking the ad, the prospect can further submit a credit application, which is counted as a success customer acquisition. Table 2 presents the data summary.

To guide the model training, we need to set the reward properly. As the bank pays ¥75 per 1,000 ad exposures across all the collaborated apps, we set the unit cost of ad exposure to ¥0.075. After consulting with our data provider, we set the average revenues of an ad click and a customer acquisition as ¥2.60 and ¥260, respectively. As the goal of the model is bringing in new customers and standard service will be applied to all customers, we do not distinguish the rewards of acquiring customers with different lifetime values. Additionally, after a prospect submits an application, the bank reviews it and decides whether to accept it or not. This decision-making process is carried out by the bank and is independent of the proposed model, which only aims to attract prospects to submit applications.

5.2. Benchmark Models and Evaluation

We randomly divided the data into a training sample (80% of the prospects) and a holdout sample (20% of the prospects). For evaluating the short-term and long-term revenues from the proposed model and benchmarks, we employ off-policy RL model evaluation. The detailed mechanism of off-policy evaluation is outlined in Online Appendix D. Specifically, we compare the performance of the proposed model with the following RL algorithms.

Table 2. Data Summary

Entities	Unique number
Static feature	
Age	52
Gender	3
Associated industry	6
City	312
Marital status	2
Dynamic feature	
Month	12
Weekday	7
Time	24
Holiday	2
Time gap	51,165
Internet connection	3
TOL	10
Device	5
App	96
Ad	195
Entities	Total number
Interaction summary	
Interaction	3,555,175
Ad exposure	1,683,180
Ad click	77,210
Customer acquisition	81

1. Contextual multiarmed bandit (Lu et al. 2010). Our partner currently adopts contextual MAB in their customer acquisition system. The model takes the inputs that combine the content features of the ad, static characteristics of the customer, and historical interactions as well as contextual features of the incoming interaction. Given a state, the objective of the model is to learn the reward distribution (Gaussian) for each action. Then, ϵ -greedy is adopted to balance exploration and exploitation to decide the action to deploy.⁸ The estimation is updated in real-time based on the feedback of the prospect. In contrast to the proposed model that aims to optimize the long-term reward, MAB is a myopic approach that focuses on immediate reward optimization (CTR).

2. DQN (Mnih et al. 2015). Here, we adopt the classical DQN that uses a deep feed-forward network to process the features of the recent interactions with a prospect to summarize the state of the prospect. Specifically, following Mnih et al. (2015), we use a deep feed-forward network to process the concatenation of the latest four interactions that the agent has encountered to generate state s_{it} . Then, the state is connected with a regular dense layer (regression) to predict Q values of executing different actions. To ensure learning stability and generalizability, we also adopt double Q-learning (Van Hasselt et al. 2016) and dueling architectures (Wang et al. 2016) in DQN. It is worth noting that such a model setup makes it identical to the RL model proposed by Wang et al. (2023) for sequential promotion problems.

3. Interpretable deep Q-network (I-DQN) (Annasamy and Sycara 2019). DQN is not interpretable. Accordingly, I-DQN was developed to incorporate an attention mechanism to make model decisions more transparent. In contrast to the proposed model, I-DQN uses CNN to process the latest game frame (we apply CNN to the latest four interactions matrix in our setting) to generate the query, whereas the keys and values are randomly generated and linked to a specific action and Q -value pair. We adopted an open-source code⁹ and trained the model on the empirical data. Following the original paper, we utilized the upper confidence bound algorithm to direct action selection after training the model.

4. DRQN (Hausknecht and Stone 2015). DRQN accommodates POMDP by processing the historical interactions using an RNN to summarize the state of the prospect. Different from the proposed model that integrates RNN within the attention mechanism to learn the state representation, DRQN utilizes the standard many-to-one structure RNN to learn state representation s_{it} directly. Then, the state is connected with a regular dense layer to predict Q values of executing different actions. To ensure learning stability and generalizability, we also adopt double Q-learning (Van

Hasselt et al. 2016) and dueling architectures (Wang et al. 2016) in DRQN.

5. Dyna-DRQN (Sutton and Barto 2018). The DQN, the DRQN, and the proposed model belong to the model-free RL family that does not explicitly model the environment. Such models require substantial sample data collected from the environment to improve the policy. The Dyna architecture revises the Q-learning framework that explicitly models the environment. It simultaneously learns from observations to model the environment and utilizes both observations and simulated data from the learned model to optimize the policy. Such adjustments make Dyna Q-learning a more sampling-efficient model.

6. Spatiotemporal attention deep recurrent Q-network (ST-ADRQN). Etchart et al. (2019) introduced an attention network between LSTM's (Long Short-Term Memory) time steps to learn the most important states from history. The output of LSTM at each time stamp is used as a state to estimate the Q values for different actions. DRQN-attention is different from ST-ADRQN as the proposed model aims to customize the attention mechanism to identify which channel is promising for the incoming interaction, whereas ST-ADRQN adopts the attention mechanism as to track back into the history to identify "important states." Additionally, ST-ADRQN utilizes the same data to construct all three components (i.e., query, key, and value) of the attention network; we use the historical state and static information as the query, whereas features in the incoming interaction are used for key and value. Finally, there is an attention network in every time step in ST-ADRQN, whereas there is only one attention network in the DRQN-attention model.

We utilize Fixed- M Per-Episode Rejection Sampling (PERS) (Mandel et al. 2016) for the off-policy evaluation, which aims to sample episodes from the test set. Fixed- M PERS ensures that the sampled episodes align with the learned policy of the RL model being evaluated. An episode with length H represents an ordered trajectory of actions, rewards, and states: $(s_1, a_1, r_1, \dots, r_H)$. For example, when $H = 3$, we sample candidate episodes as $\{s_1, a_1, r_1, s_2, a_2, r_2, s_3, a_3, r_3\}$. For all the sampled episodes, we calculate the average reward per episode to evaluate model performance. More details of off-policy evaluation can be found in Online Appendix D. Table 3 compares the average rewards of sampled episodes via the off-policy evaluation with different lengths ($H = 1, 3, 6$). We have the following observations. (1) MAB performs better than all Q-learning models (DQN, DRQN, Dyna-DRQN, and DRQN-attention) when we set $H = 1$, but Q-learning models surpass MAB with longer episodes $H = 3, 6$. Such a comparison shows that MAB is indeed superior at learning intervention strategies to optimize immediate reward, but Q-learning models will learn strategical policies that better balance immediate reward and long-term rewards to achieve greater overall

Table 3. Average Reward Comparison Across Different Models and Different Episode Lengths

Algorithm	AR@1	AR@3	AR@6
MAB	0.0568	0.1721	0.3406
DQN	0.0397	0.1939	0.3856
I-DQN	0.0351	0.1836	0.3789
DRQN	0.0450	0.1983	0.3993
ST-ADRQN	0.0382	0.1667	0.3628
Dyna-DRQN	0.0379	0.1621	0.3507
DRQN-attention	0.0412	0.1990	0.3999

Note. All average rewards (ARs) are statistically significantly different from those of MAB at the 1% significance level.

rewards. (2) DRQN and DRQN-attention lead to higher rewards than the vanilla DQN. Their superior performances confirm the importance of incorporating more historical interactions and the necessity of using RNN to accommodate POMDP. (3) Compared with DRQN, the attention mechanism in DRQN-attention does not sacrifice performance, as their performances are not statistically significantly different ($p = 0.708$). However, incorporating the attention mechanism helps explain the decision process of the model by emphasizing those task-relevant features of the incoming interaction (Section 6). (4) I-DQN also adopts an attention mechanism, but the design of the attention mechanism makes it not perform as well as the proposed DRQN-attention. The key and value components in I-DQN are randomly generated and specific to an action and Q -value pair, serving to identify states where the agent expects to receive a specific Q value by taking the given action. Because of the action space being much larger in our setting (e.g., 195 ads in our setting versus five actions in the spaceinvader game used in Annasamy and Sycara 2019), such a treatment contains excessive noisy information as the key and value are randomly generated to each action and Q -value pair. Additionally, only using the latest interactions to learn the query might not accommodate POMDP. Thus, such attention mechanism design leads to inferior model performance in the customer acquisition setting. (5) ST-ADRQN also adopts attention networks into DRQN. However, DRQN-attention only has one attention module that processes the output of RNN, but ST-ADRQN has an attention module at each time step within RNN. Additionally, ST-ADRQN uses the same data to learn query, key, and value in the attention network. Such a model setting makes ST-ADRQN more complex but less informative than DRQN-attention. Beyond the testing performance in Table 3, we found that ST-ADRQN was not fitting well in the training set as well. Such observation is consistent with the finding of ST-ADRQN performance in the fully observable Pong game (Etchart et al. 2019). This is explained by He et al. (2016). The deeper and more complex neural networks are not always necessary to perform better than their simpler counterparts; the main reason is that such

models are harder to train. (6) Finally, Dyna-DRQN does not perform as well as other models, potentially because using simulated data introduces noise that undermines performance.

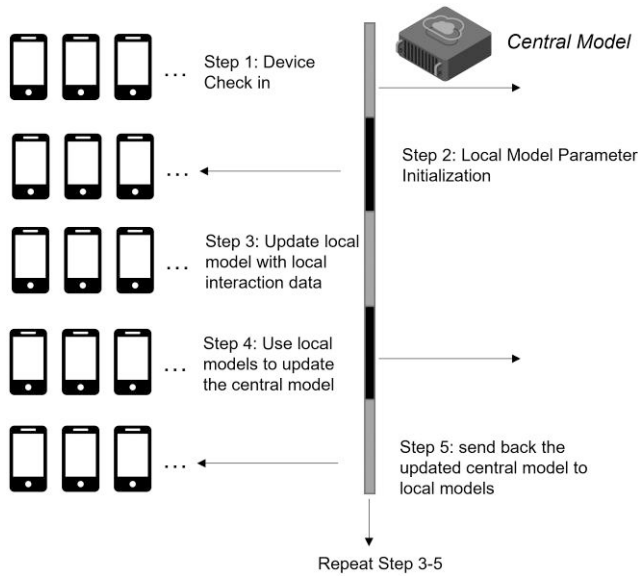
Furthermore, by comparing the average reward per interaction across different H values, we find out that the improvement in the performance of DRQN-attention over MAB is not linear. When $H = 3$, the improvement of average reward per interaction of DRQN-attention over MAB is ¥0.0089 ($\approx (0.1990 - 0.1721)/3$). With $H = 6$, such improvement increases to ¥0.0098 ($\approx (0.3999 - 0.3406)/6$). This comparison further confirms that the RL model aims to optimize the long-term reward rather than the immediate reward. To sum up, DRQN-attention outperforms the company's current policy and other RL algorithms, showing its effectiveness in customer acquisition.

5.3. Federated RL for Privacy-Sensitive Context

Customers are becoming increasingly aware and protective of their data. With the rise of regulations, such as General Data Protection Regulation, and the initiatives from iOS and Android platforms, protecting user data privacy has become a widespread global trend (Kollnig et al. 2021).

Correspondingly, we should consider preventing personal data leakage and protecting personal privacy when applying the DRQN-attention model. Privacy-enhancing technologies (PETs) have recently been identified as a pivotal area for future research in the context of privacy regulation because they offer a promising solution to the ongoing tension among privacy concerns, customer welfare, and the development of a data-driven economy (Johnson 2022). The Information Commissioner's Office of the UK Government defines PETs as "technologies that embody fundamental data protection principles by minimizing personal data use, maximizing data security, and empowering individuals" (ICO 2022). FL is one of the notable examples of PETs. Thus, we adopt federated learning to distribute the model training and deployment. Federated learning enables collaboration between agents without exchanging raw data, ensuring the privacy of sensitive information. This integration of federated learning and RL has led to the research area of federated RL (Qi et al. 2021). Specifically, we adopt the federated RL policy from Tehrani et al. (2021) to learn and deploy RL models. The federated approach allows us to train the model while keeping personal data stored and processed locally on local devices, with only intermediate model updates being communicated to a central model. The central model then aggregates the updates and sends a refined model back to the devices to support sequential targeting decisions. The federated learning process is illustrated in Figure 5. We evaluate the performance of the federated version of RL models using the same training and testing set. The result, shown in Table 4, indicates that the federated versions have a

Figure 5. Federated Learning of RL Models



similar performance to the original version (Table 3), with only a marginal drop. This confirms that the model can be applied in privacy-sensitive settings without compromising the service quality.¹⁰

6. What Do Attention Weights Capture?

Although the attention map may not encapsulate the entire decision-making process, it provides valuable insights into the strategies learned by the model. In this section, we explore what attention weights capture.

For each interaction, the attention weights, denoted as att_{it} , interact with v_{it} (a copy of f_{it}) in Equation (6) through element-wise multiplication. The outcome of Equation (6) is then used to estimate the state s_{it} in Equation (7), influencing subsequent model decisions. Therefore, the high-weighted elements in att_{it} significantly impact state construction and consequently, the model's decisions. Given that f_{it} is a one-hot encoding vector, each element in f_{it} is binary and represents a feature in the incoming interaction. For instance, if the first element in f_{it} is equal to one, it indicates that the incoming interaction is from app 1; similarly, if the 97th element is equal to one, it denotes that the interaction occurred on Monday. Consequently, we are interested

in the high-weight element in the attention weights att_{it} and whether the corresponding element in f_{it} is filled with one. Specifically, within each att_{it} , we identify the elements with the top 10 attention weights. We then categorize each interaction into two groups: (1) “matched with attention weight” if there are greater than or equal to five corresponding elements in f_{it} equal to one and (2) “unmatched with attention weight” if there are less than or equal to two corresponding elements in f_{it} equal to one. In Table 5 (the first two rows), we observe that across episodes with different lengths, the average rewards of episodes in the “matched with attention weight” group consistently exceed those in the “unmatched” group. This suggests that attention weight may be aimed at identifying features in the incoming session that lead to higher rewards.

However, if we are only interested in identifying those features of the incoming interaction that lead to higher rewards, there are other simpler options, such as the association rule (Piatetsky-Shapiro 1991). The association rule intends to identify strong rules in the data set using measures such as *confidence*. The confidence value of a rule, $X \Rightarrow Y$, is defined as among all transactions starting from X , the proportion of the transactions containing Y . Specifically, we may define static characteristics and historical interactions as X and features of the incoming interaction as Y , and we may calculate the confidence values from all consecutive observations. We restrict this analysis to the incoming interactions that lead to positive rewards (e.g., click or customer acquisition). Therefore, the association rules could also capture which features of the incoming interactions are more likely to lead to higher rewards. In fact, in our data, the attention weights and the weights of features in Y based on the association rule have a high correlation of 0.8412. Similar to how we assign an incoming interaction into the “matched with attention weight” group, if an interaction contains greater than or equal to five features with top 10 confidence values of the association rule, we assign it to the “matched with association rule” group. In Table 5, we further compare the average rewards of episodes “matched with association rule” against those “matched with attention weight” (row 3 and row 1, respectively). With the exception of AR@1, episodes “matched with association rule” consistently result in lower average rewards. The reason behind the attention weight having higher average rewards is that

Table 4. Average Reward Comparison Across RL Models via Federated Learning

Algorithm	AR@1	AR@3	AR@6
DQN	0.0388	0.1806	0.3809
I-DQN	0.0339	0.1802	0.3735
DRQN	0.0437	0.1956	0.3950
ST-ADRQN	0.0368	0.1619	0.3513
DRQN-attention	0.0403	0.1958	0.3892

Note. AR, average reward.

Table 5. Average Reward Comparison Between Attention Mechanism and Association Rule

Group	AR@1	AR@3	AR@6
Matched with attention weight	0.0417	0.2298	0.4815
Unmatched with attention weight	0.0218	0.0845	0.1738
Matched with association rule	0.0433	0.1939	0.3946

Note. AR, average reward.

the association rule only focuses on immediate outcomes of the incoming interaction with prospects, whereas the attention mechanism considers optimal trajectories that lead to higher long-term rewards. Accordingly, the “matched with association rule” group has a higher average reward with AR@1 but is dominated by the “matched with attention weight” group with longer episodes. Such a comparison shows that the attention mechanism spots those features in the incoming interaction that aim for optimal long-term reward. Consequently, the model learns to scan through possible future paths and decides on optimal ones with promising outcomes.¹¹

7. Attention Weight and Advertising Channels

In this section, we explore how the attention mechanism chooses advertising channels, aiming to obtain a better understanding of the decision-making process of the proposed model. Using three illustrative cases, we demonstrate how the attention mechanism plans the optimal path for ad exposures. In particular, we analyze how the attention mechanism adjusts its advertising channel choices across individual prospects and over time. The three specific cases include (1) advertising channel choice for prospects from different industries, (2) advertising channel adjustment according to dynamic prospect behaviors, and (3) marketing channel planning that accounts for the seasonality of the agricultural industry. To facilitate these analyses, we first define two new concepts. The first is most important features (*MIFs*). For each att_{it} , we call the corresponding 10 features in f_{it} with the highest weights in att_{it} as top 10 features. Across all att_{it} , some features frequently appear as one of the top 10 features. We select features whose frequencies of being in the top 10 lists are above the 80 percentile and label them as *MIF*. In total, there are 26 *MIFs*. *MIFs* can be viewed as features that are important in general across all att_{it} . (2) The second is singular features. Given an att_{it} , any of its top 10 features that are not within *MIF* will be considered as a singular feature. Singular features can be viewed as how the attention weights of an interaction deviate from the general pattern.

7.1. Case 1: Advertising Channel Choice Based on Associated Industry

In this first case, we collected all the interactions associated with groups of prospects working in specific industries, including information technology (IT), finance and law, and education. For interactions associated with each group of these prospects, we extracted the top 10 features based on the average attention weight footnote those singular features in Table 6. First, we noticed that prospects from the IT and finance and law industries share many similarities as singular features, like “App: FianceNews1” (an app for financial news) and “TOL: Office Building” (locations like office buildings), are within the top 10 feature list for both. This makes sense as most of them work in office buildings and are interested in financial news to manage their businesses. However, prospects from the IT industry have their specific singular features, like “App: TechNews1” (an app for technology news) and “TOL: Company & Factory” (locations like companies and factories), whereas prospects from finance and law have their special singular feature of “TOL: Finance & Insurance” (locations like banks and financial institutions). Such comparison shows that the model can learn to target prospects from different industries based on their unique characteristics. When we turn to teachers in the education industry, the model learns completely different targeting patterns. The singular features of “App: News3” (an app for general news) and “TOL: Education & Research” (schools and businesses in the education industry) appear in the top 10 list. Thus, we understand that technology news and financial news are less promising channels to target teachers compared with general news. Additionally, teachers are more likely to be located in schools or educational institutes rather than office buildings.

7.2. Case 2: Advertising Channel Adjustment Because of Dynamic Customer Behaviors

“App: LiveStream1” (an app for streaming live sporting events and dramas) is the fifth *MIF* and a very popular app for targeting customers. In Table 7, for case 2.1, we collect those interactions with prospects whose previous interactions include at least five ad exposures on this

Table 6. Attention Weight Based on Associated Industry

	IT	Finance and law	Education
Top 1	Time Gap	Time Gap	Time Gap
Top 2	App: LiveStream1	Connection: Wi-Fi	Monday
Top 3	Device: Cellphone	App: LiveStream1	Tuesday
Top 4	Monday	TOL: Finance & Insurance ^a	Connection: Wi-Fi
Top 5	App: TechNews1 ^a	Device: Cellphone	App: System2
Top 6	App: FianceNews1 ^a	Cumulative Impression	App: E-Reading1
Top 7	TOL: Office Building ^a	TOL: Office Building ^a	App: News3 ^a
Top 8	Cumulative Impression	App: FianceNews1 ^a	App: Browser1
Top 9	TOL: Company & Factory ^a	Connection: 4G	Device: Cellphone
Top 10	Connection: Wi-Fi	App: Browser1	TOL: Education & Research ^a

^aSingular feature.

Table 7. Attention Weight Adjusted to Different Customer Behaviors

	Case 2.1	Case 2.2
Top 1	Time Gap	Time Gap
Top 2	Device: Cellphone	App: LiveStream1
Top 3	AD Platform3	App: CCTV1 ^a
Top 4	Wednesday	Device: Cellphone
Top 5	App: E-Reading1	App: Trans1 ^a
Top 6	Cumulative Impression	Cumulative Impression
Top 8	Connection: UK	Wednesday
Top 9	App: Browser2	App: Browser2
Top 10	App: System2	Connection: UK

^aSingular feature.

app, but no ad click or customer acquisition occurred. We then find the top 10 features for case 1 by averaging the attention weights in this group. We observe that “App: LiveStream1” is not one of the top 10 features, even though it is the fifth *MIF*. The disappearance of “App: LiveStream1” indicates that the model is able to automatically adjust to avoid ineffective channels. For case 2.2 in the same table, we turn our attention to “App: CCTV1” (a closed-circuit television monitoring app), an app not being one of the *MIFs*. We collect those interactions of prospects whose previous interactions include one or more ad clicks on this app. For the subsequent interactions with this group of prospects, the top 10 features include two singular features: “App: CCTV1” and “App: Trans1” (an app for language translation). It is unsurprising to see that “App: CCTV1” is being emphasized. However, why is “App: Trans1” also being promoted? From the data, we find out that the probability of an ad click on “App: Trans1” conditional on a previous ad click on “App: CCTV1” is 23.82%, substantially higher than the average rate in the data ($< 0.01\%$). The appearance of “App: Trans1” as one top 10 feature indicates that the model can learn to target prospects across apps over time for better long-term rewards. Such insight could help managers better understand the coordination of different ad channels for customer acquisition.

7.3. Case 3: Advertising Channel Seasonality for Targeting Agricultural Entrepreneurs

In this exercise, we analyze how the proposed model learns to target prospects who are identified as agricultural entrepreneurs by the bank. Major agricultural loans in China are seasonal (Agricultural Bank of China 2022), and they are used to finance grain purchase, storage, and transportation during the peak harvest seasons of agricultural products. Because of the seasonality of such loans, it is interesting to see how the model behaves during different periods of the year. In the second half of 2019 (the duration of our data), July to September and September to December are the peak seasons for

summer and fall grain purchases, respectively (LSWZ 2022). Thus, we segment the second half of 2019 into two stages: (1) prepeak stage: June and (2) peak stage: July to December. The top 10 features based on the average attention weight during prepeak and peak seasons for agricultural entrepreneurs are shown in Table 8. Singular features are highlighted. It is not surprising to see non-*MIF* apps “App: Weather1” (a weather forecasting app) and “App: CCTV1” are among the top 10 lists because they are essential apps for the agricultural industry and hence, effective channels to target agricultural entrepreneurs. Moreover, non-*MIF* apps “App: E-Commerce1” and “App: LiveStream2” become promising advertising channels and appear in the top 10 list during the prepeak stage, indicating that the potential customers lean toward consumption and entertainment before the peak season. Thus, showing ads in these channels could influence potential customers by priming them with the bank as a future lender. Once the peak season arrives, non-*MIF* “TOL: Finance & Insurance” and “App: FinanceNews1” become prominent on the top 10 list. They are strong signals as visiting a brick-and-mortar finance and insurance facility or browsing finance information online indicates that the user might need financial services. Thus, the model learns to precisely target the prospects to meet their needs.

The agricultural industry exhibits different seasonality patterns than other industries, and these nuanced insights in case 3 are typically overlooked or are not precisely captured by marketing managers. They provided valuable assistance to our data provider on two fronts. The first is industry trends monitoring. Agricultural industry clients were not traditionally part of our data provider’s clientele. However, the RL model continuously updated its targeting policy through continuous interactions with prospects, signaling emerging patterns in the agricultural industry. This enabled managers to recognize the needs within this sector and its unique funding patterns. Consequently, they engaged with agricultural industry associations before the peak season to disseminate loan information and establish

Table 8. Attention Weight to Target Agricultural Entrepreneurs at Two Stages

	Prepeak	Peak
Top 1	Time Gap	Time Gap
Top 2	App: Weather1 ^a	App: Weather1 ^a
Top 3	AD Platform3	App: News1
Top 4	App: News1	AD Platform3
Top 5	App: CCTV1 ^a	TOL: Finance & Insurance ^a
Top 6	Cumulative Impression	App: CCTV1 ^a
Top 8	App: LiveStream2 ^a	Cumulative Impression
Top 9	App: Browser2	App: FinanceNews1 ^a
Top 10	App: E-Commerce1 ^a	TOL: Company & Factory ^a

^aSingular feature.

dedicated channels to meet these needs during peak seasons. Thus, the insights helped monitor industry trends and transformed business-to-consumer transactions into business-to-business opportunities. The second is customized product design. After identifying the seasonality patterns in the agricultural industry, our data provider introduced a customized financial product specifically tailored for agricultural entrepreneurs. Although other industries typically have steady cash flows, grain purchases/sales in agriculture occur only during short periods with intense transactions, with long idle periods in between. Consequently, the customized product was designed to align with the seasonality of the agricultural industry, adopting a more flexible repayment mode instead of regular monthly installments.

Allen et al. (2023) discussed practical validation strategies for interpretations from explainable machine learning models. A key validation strategy involves randomly splitting the available data into a training set and a test set and validating the findings in the testing set. The fundamental idea is to utilize explainable machine learning models on the training data to generate interpretations or insights. Then, one can evaluate the model's insights on the test data. In each case, we aim to conclude that the specific advertising channels in each case could lead to more promising outcomes (e.g., higher long-term rewards). Thus, we could verify in the testing data whether such insights and conclusions are valid. We employ the same validation strategy as in Section 6, which compares the attention weight with association rules. For each case in this section, we collect relevant data from the testing set. For example, we gather all data associated with the occupation of teachers. We then split these data into two sets. The first is matched, where the prospect landed on one of the recommended advertising channels and an ad was shown to the prospect. The second is unmatched, where the prospect did not land on any of the recommended advertising channels or an ad was not shown to the prospect in the recommended advertising channel. We also compare the average reward at one, three, and six of these two groups in all three cases in this section. As shown in Table 9, the average reward in the matched group is higher than that of the unmatched group. Such a comparison validates the insights gained from the proposed model on the testing set, proving that the learned insights from the three cases are valid.

The presented cases illustrate that the proposed model can effectively learn optimal advertising channels for diverse populations, adapt to dynamic customer behaviors, and grasp the seasonality of the industry, allowing for corresponding adjustments in advertising channels. A comparison with a post hoc explainable model is also conducted to highlight differences in explainability with the proposed intrinsic explainable model, detailed in Online Appendix G.

Table 9. Average Reward Between Matched/Unmatched Groups with Recommend Advertising Channels

Case	Group	AR@1	AR@3	AR@6
Industry: IT	Matched	0.0308	0.1037	0.2263
	Unmatched	0.0202	0.0805	0.1705
Industry: Finance & law	Matched	0.0284	0.0935	0.1866
	Unmatched	0.0213	0.0783	0.1760
Industry: Education	Matched	0.0253	0.0854	0.1580
	Unmatched	0.0206	0.0732	0.1532
Case 2.1	Matched	0.0262	0.0807	0.1538
	Unmatched	0.0217	0.0683	0.1361
Case 2.2	Matched	0.0382	0.2033	0.4539
	Unmatched	0.0191	0.0862	0.1701
Industry: Agriculture (prepeak)	Matched	0.0306	0.0583	0.1367
	Unmatched	0.0211	0.0517	0.1233
Industry: Agriculture (peak)	Matched	0.0736	0.2368	0.4236
	Unmatched	0.0226	0.0757	0.1321

Note. AR, average reward.

8. Conclusion

We introduce the DRQN-attention model that aims to optimize the long-term reward for customer acquisition and meanwhile, provide meaningful explanations for the model decisions. Our empirical analyses show that the proposed model results in higher long-term revenues than state-of-the-art RL models. Furthermore, we provide evidence that DRQN-attention is able to pinpoint important features of targeting opportunities and guide advertising channel choices while presenting intuitive explanations for the decisions of the RL model. In short, the proposed model has good explainability and applicability, and thus, it can be applied widely for customer acquisition. This study provides a valuable tool for digital businesses seeking to improve their customer acquisition efforts. By leveraging sequential messaging through RL, the proposed model can optimize customer acquisition and drive potential customers down the acquisition funnel.

We acknowledge certain limitations in our current study. Our proposed model excels in learning policies to optimize long-term rewards and reveals attention weights indicating how the model targets different prospects. However, the attention weight itself is not directly interpretable like the results of the decision tree, and extra effects, like grouping based on static information or dynamic information of prospects and comparing with the most important features, are needed to learn the pattern of targeting different prospects. Future research avenues could explore methods for directly learning such patterns from the data. Moreover, it is essential to note that the empirical data in our study may not comprehensively cover the whole state-action space. Consequently, the learned policy in our reinforcement learning model may not be globally optimal. If this model gets a chance to run in the on-policy environment, potential enhancements include employing

Boltzmann exploration (Bertsekas and Tsitsiklis 1995) or other exploration approaches to refine the learned policy and update insights into targeting different prospect types. Additionally, the RL model learns to target prospects based on their static and dynamic features. A promising future direction involves incorporating a fairness perspective into the proposed model. This enhancement aims to ensure equitable delivery of marketing messages across prospects while still maintaining the goal of optimizing long-term rewards. Lastly, the proposed DRQN-attention model presents a versatile solution designed to learn optimal sequential intervention policies for user/customer acquisition. Importantly, this model is not inherently reliant on crossapp tracking. Its applicability extends to various contexts, such as within an app, where it can effectively deliver tailored sequential messages for each interaction within the respective platform.

Endnotes

- ¹ Throughout the paper, we use the terms “reward” and “revenue” interchangeably as they convey the same meaning in our context.
- ² MAB is an RL model with a single state, where the following observation revealed by the environment to the agent is not influenced by the preceding actions (Sutton and Barto 2018).
- ³ This includes models such as decision trees, linear models, and attention models among others.
- ⁴ Prospect’s arrival on the collaborated channel is an exogenous process.
- ⁵ The reward of customer acquisition should be larger than the reward of ad click. We will specify the reward setting in Section 5.1.
- ⁶ The detailed structure of the hi_t vector is specified in Online Appendix B. Prospect reaction is the 1/0 code to represent the click/unclick of the ad in the historical interaction. However, the prospect reaction to the incoming t th interaction is unknown, and we set it to -1 .
- ⁷ The data collection is in compliance with Personal Information Protection Law of the People’s Republic of China.
- ⁸ We could not disclose the setting of epsilon because of an Non-Disclosure Agreement. By exploring the data generated by contextual MAB in Online Appendix E, we find that the contextual MAB does try different actions under different states to introduce action randomness in the exploration process.
- ⁹ The code is at <https://github.com/maraghuram/I-DQN>.
- ¹⁰ Online Appendix H discuss how to enhance the security and privacy of FL models to different attacks.
- ¹¹ To validate the explainability of model decisions using attention weights, we follow the attention weight validation procedure outlined by Serrano and Smith (2019). The results indicate that the high-weighted features indeed play a crucial role in generating model decisions. Additional details can be found in Online Appendix F.

References

- Agricultural Bank of China (2022) Agriculture seasonal loan. Accessed July 10, 2023, <https://www.abchina.com/cn/RuralSvc/Businesses/jjdk/>.
- Allen GI, Gan L, Zheng L (2023) Interpretable machine learning for discovery: Statistical challenges and opportunities. *Annual Rev. Statist. Appl.* 11:97–121.
- Annasamy RM, Sycara K (2019) Toward better interpretability in deep Q-networks. *Proc. Conf. AAAI Artificial Intelligence* 33(01):4561–4569.
- AppsFlyer (2021) Fintech marketers invested \$3B on user acquisition in 2020 according to AppsFlyer. Accessed July 10, 2023, <https://www.bloomberg.com/press-releases/2021-06-09/fintech-marketers-invested-3b-on-user-acquisition-in-2020-according-to-appsflyer>.
- Berente N, Gu B, Recker J, Santhanam R (2021) Managing artificial intelligence. *Management Inform. Systems Quart.* 45(3):1433–1450.
- Bertsekas DP, Tsitsiklis JN (1995) Neuro-dynamic programming: An overview. *Proc. 1995 34th IEEE Conf. Decision Control*, vol. 1 (IEEE, Piscataway, NJ), 560–564.
- Brauwiers G, Frasincar F (2021) A general survey on attention mechanisms in deep learning. *IEEE Trans. Knowledge Data Engrg.* 35(4):3279–3298.
- Cao J, Leng Y (2021) Adaptive data acquisition for personalized recommender systems with optimality guarantees on short-form video platforms. Preprint, submitted May 11, <http://dx.doi.org/10.2139/ssrn.3843086>.
- Chaudhari S, Mithal V, Polatkan G, Ramanath R (2021) An attentive survey of attention models. *ACM Trans. Intelligent Systems Tech.* 12(5):1–32.
- Chuck L (2022) The four pillars of customer acquisition strategy. *Forbes* (August 15), <https://www.forbes.com/sites/forbesbusinesscouncil/2022/08/15/the-four-pillars-of-customer-acquisition-strategy/?sh=a84f6f63218d>.
- Du M, Liu N, Hu X (2019) Techniques for interpretable machine learning. *Comm. ACM* 63(1):68–77.
- Etchart M, Ladosz P, Mulvaney D (2019) Spatio-temporal attention deep recurrent Q-network for POMDPs. *Proc. Progress Artificial Intelligence 19th EPIA Conf. Artificial Intelligence EPIA 2019* (Springer, Cham, Switzerland), 98–105.
- Fei H, Zhang Y, Ren Y, Ji D (2021) Optimizing attention for sequence modeling via reinforcement learning. *IEEE Trans. Neural Networks Learn. Systems* 33(8):3612–3621.
- Fernández-Loría C, Provost F (2022) Causal decision making and causal effect estimation are not the same ... and why it matters. *INFORMS J. Data Sci.* 1(1):4–16.
- Frick TW, Belo R, Telang R (2023) Incentive misalignments in programmatic advertising: Evidence from a randomized field experiment. *Management Sci.* 69(3):1665–1686.
- Hauser JR, Urban GL, Liberali G, Braun M (2009) Website morphing. *Marketing Sci.* 28(2):202–223.
- Hausknecht M, Stone P (2015) Deep recurrent Q-learning for partially observable MDPs. 2015 AAAI Fall Sympos. Series (AAAI Press, Palo Alto, CA).
- He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. *Proc. IEEE Conf. Comput. Vision Pattern Recognition* (IEEE, Piscataway, NJ), 770–778.
- Hoffman DL, Novak TP (2000) How to acquire customers on the web. *Harvard Bus. Rev.* (May–June), <https://hbr.org/2000/05/how-to-acquire-customers-on-the-web>.
- ICO (2022) Draft anonymisation, pseudonymization and privacy enhancing technologies guidance. (December 31), <https://ico.org.uk/about-the-ico/ico-and-stakeholder-consultations/ico-call-for-views-anonymisation-pseudonymisation-and-privacy-enhancing-technologies-guidance/>.
- Itaya H, Hirakawa T, Yamashita T, Fujiyoshi H, Sugiura K (2021) Visual explanation using attention mechanism in actor-critic-based deep reinforcement learning. 2021 Internat. Joint Conf. Neural Networks (IJCNN) (IEEE, Piscataway, NJ), 1–10.
- Johnson G (2022) Economic research on privacy regulation: Lessons from the GDPR and beyond. Preprint, submitted December 8, <http://dx.doi.org/10.2139/ssrn.4290849>.
- Khanna M, Wimpey JS, Bruhn M, Singh S, Hommes M, Sorokina A (2017) MSME Finance Gap: Assessment of the Shortfalls and Opportunities in Financing Micro, Small, and Medium Enterprises in Emerging Markets (World Bank, Washington, DC).

- Kiran BR, Sobh I, Talpaert V, Mannion P, Al Sallab AA, Yogamani S, Pérez P (2022) Deep reinforcement learning for autonomous driving: A survey. *IEEE Trans. Intelligent Transportation Systems* 23(6):4909–4926.
- Kokkodis M, Ipeirotis PG (2021) Demand-aware career path recommendations: A reinforcement learning approach. *Management Sci.* 67(7):4362–4383.
- Kollnig K, Binns R, Van Kleek M, Lyngs U, Zhao J, Tinsman C, Shadbolt N (2021) Before and after GDPR: Tracking in mobile apps. *Internet Policy Rev.* 10(4).
- Leng Y, Ruiz R, Liu X (2020) Geometric deep learning based recommender system and an interpretable decision support system. Preprint, submitted November 13, <http://dx.doi.org/10.2139/ssrn.3696092>.
- LSWZ (2022) Grain circulation data, national food and strategic reserves administration. Accessed July 10, 2023, <http://www.lswz.gov.cn/html/ywpd/lstk/tj-sgsj.shtml>.
- Lu T, Pál D, Pál M (2010) Contextual multi-armed bandits. *Proc. Thirteenth Internat. Conf. Artificial Intelligence Statist.* (PMLR, New York), 485–492.
- Mandel T, Liu YE, Brunskill E, Popović Z (2016) Offline evaluation of online reinforcement learning algorithms. *Proc. AAAI Conf. Artificial Intelligence*, vol. 30 (AAAI Press, Palo Alto, CA).
- Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, Graves A, et al. (2015) Human-level control through deep reinforcement learning. *Nature* 518(7540):529–533.
- Mott A, Zoran D, Chrzanowski M, Wierstra D, Jimenez Rezende D (2019) Toward interpretable reinforcement learning using attention augmented agents. *Adv. Neural Inform. Processing Systems* 32(1107):12360–12369.
- Piatetsky-Shapiro G (1991) Discovery, analysis, and presentation of strong rules. Piatetsky-Shapiro G, Frawley W, eds. *Knowledge Discovery in Databases* (MIT Press, Cambridge, MA), 229–248.
- Puiutta E, Veith EM (2020) Explainable reinforcement learning: A survey. *Internat. Cross-Domain Conf. Machine Learn. Knowledge Extraction* (Springer, Cham, Switzerland), 77–95.
- Qi J, Zhou Q, Lei L, Zheng K (2021) Federated reinforcement learning: Techniques, applications, and open challenges. Preprint, submitted, August 26, <https://arxiv.org/abs/2108.11887>.
- Reiley D, Lewis RA, Schreiner T (2012) Ad attributes and attribution: Large-scale field experiments measure online customer acquisition. Preprint, submitted May 2, <http://dx.doi.org/10.2139/ssrn.2049457>.
- Schwartz EM, Bradlow ET, Fader PS (2017) Customer acquisition via display advertising using multi-armed bandit experiments. *Marketing Sci.* 36(4):500–522.
- Serrano S, Smith NA (2019) Is attention interpretable? Korhonen A, Traum D, Márquez L, eds. *Proc. 57th Annual Meeting Assoc. Comput. Linguistics* (Association for Computing Machinery, New York), 2931–2951.
- Shi W, Huang G, Song S, Wang Z, Lin T, Wu C (2020) Self-supervised discovering of interpretable features for reinforcement learning. *IEEE Trans. Pattern Anal. Machine Intelligence* 44(5):2712–2724.
- Simester D, Timoshenko A, Zoumpoulis SI (2020) Targeting prospective customers: Robustness of machine-learning methods to typical data challenges. *Management Sci.* 66(6):2495–2522.
- Song Y, Sun T (2023) Ensemble experiments to optimize interventions along the customer journey: A reinforcement learning approach. *Management Sci.*, ePub ahead of print October 4, <https://doi.org/10.1287/mnsc.2023.4914>.
- Song Y, Sahoo N, Srinivasan S, Dellarocas C (2022) Uncovering characteristic response paths of a population. *INFORMS J. Comput.* 34(3):1661–1680.
- Sutton RS, Barto AG (2018) *Reinforcement Learning: An Introduction* (MIT Press, Cambridge, MA).
- Tehrani P, Restuccia F, Levorato M (2021) Federated deep reinforcement learning for the distributed control of NextG wireless networks. 2021 *IEEE Internat. Sympos. Dynamic Spectrum Access Networks (DySPAN)* (IEEE, Piscataway, NJ), 248–253.
- Torrado RR, Bontrager P, Togelius J, Liu J, Perez-Liebana D (2018) Deep reinforcement learning for general video game AI. 2018 *IEEE Conf. Comput. Intelligence Games (CIG)* (IEEE, Piscataway, NJ), 1–8.
- Turek M (2018) Explainable artificial intelligence. *Defense Advanced Research Projects Agency*. Accessed July 10, 2023, <https://www.darpa.mil/program/explainable-artificial-intelligence>.
- Van Hasselt H, Guez A, Silver D (2016) Deep reinforcement learning with double Q-learning. *Proc. AAAI Conf. Artificial Intelligence*, vol. 30 (AAAI Press, Palo Alto, CA).
- Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. *NIPS* 2017:5998–6008.
- Wang T, He C, Jin F, Hu YJ (2022) Evaluating the effectiveness of marketing campaigns for malls using a novel interpretable machine learning model. *Inform. Systems Res.* 33(2): 659–677.
- Wang W, Li B, Luo X, Wang X (2023) Deep reinforcement learning for sequential targeting. *Management Sci.* 69(9):5439–5460.
- Wang J, Zhang Y, Tang K, Wu J, Xiong Z (2019) Alphastock: A buying-winners-and-selling-losers investment strategy using interpretable deep reinforcement attention networks. *Proc. 25th ACM SIGKDD Internat. Conf. Knowledge Discovery Data Mining* (Association for Computing Machinery, New York), 1900–1908.
- Wang Z, Schaul T, Hessel M, Hasselt H, Lanctot M, Freitas N (2016) Dueling network architectures for deep reinforcement learning. *Internat. Conf. Machine Learn.* (PMLR, New York), 1995–2003.
- World Bank (2019) Small and medium enterprises (SMEs) finance. *World Bank* (October 16), <https://www.worldbank.org/en/topic/sme/finance>.
- Watkins CJ, Dayan P (1992) Q-learning. *Machine Learn.* 8(3–4):279–292.
- Zhang J, Curley SP (2018) Exploring explanation effects on consumers' trust in online recommender agents. *Internat. J. Human Comput. Interaction* 34(5):421–432.
- Zhang Q, Du Q, Liu G (2021) A whole-process interpretable and multi-modal deep reinforcement learning for diagnosis and analysis of Alzheimer's disease. *J. Neural Engrg.* 18(6):066032.