Communicative MARL-based Relevance Discerning Network for Repetition-Aware Recommendation

Kaiyuan Li
Beijing University of Posts and Telecommunications, Beijing, China and Meituan, Beijing, China
tsotfsk@bupt.edu.cn

Pengfei Wang*
Beijing University of Posts and Telecommunications, Beijing, China and Key Laboratory of Interactive Technology and Experience System, Ministry of Culture and Tourism, Beijing, China
wangpengfei@bupt.edu.cn

Haitao Wang
Meituan, Beijing, China
wanghaitao13@meituan.com

Qiang Liu
Meituan, Beijing, China
liuqiang43@meituan.com

Xingxing Wang
Meituan, Beijing, China
wangxingxing04@meituan.com

Dong Wang
Meituan, Beijing, China
wangdong07@meituan.com

Shangguang Wang
Beijing University of Posts and Telecommunications, Beijing, China
sgwang@bupt.edu.cn

ABSTRACT

The repeated user-item interaction now is becoming a common phenomenon in the e-commerce scenario. Due to its potential economic profit, various models are emerging to predict which item will be re-interacted based on the user-item interactions. In this specific scenario, item relevance is a critical factor that needs to be concerned, which tends to have different effects on the succeeding re-interacted one (i.e., stimulating or delaying its emergence). It is necessary to make a detailed discernment of item relevance for a better repetition-aware recommendation. Unfortunately, existing works usually mixed all these types, which may disturb the learning process and result in poor performance.

In this paper, we introduce a novel **C**ommunicative M**A**RL-based **R**elevance **D**iscerning Network (**CARD** for short) to automatically discern the item relevance for a better repetition-aware recommendation. Specifically, **CARD** formalizes the item relevance discerning problem into a communication selection process in MARL. **CARD** treats each unique interacted item as an agent and defines three different communication types over agents, which are stimulative, inhibitive, and noisy respectively. After this, **CARD** utilizes a Gumbel-enhanced classifier to distinguish the communication types among agents, and an attention-based Reactive Point Process is further designed to transmit the well-discerned stimulative and inhibitive incentives separately among all agents to make an effective collaboration for repetition decisions. Experimental results on two real-world e-commerce datasets show that our proposed method outperforms the state-of-the-art recommendation methods in terms of both sequential and repetition-aware recommenders. Furthermore, **CARD** is also deployed in the online sponsored search advertising system in Meituan, obtaining a performance improvement of over 1.5% and 1.2% in CTR and effective Cost Per Mille (eCPM) respectively, which is significant to the business.

CCS CONCEPTS

• **Information systems** → **Recommender systems**.

KEYWORDS

Repetition-aware Recommendation; Communication in Multi-agent Reinforcement Learning; Communication Selection; Reactive Point Process

ACM Reference Format:
Kaiyuan Li, Pengfei Wang, Haitao Wang, Qiang Liu, Xingxing Wang, Dong Wang, and Shangguang Wang. 2023. Communicative MARL-based Relevance Discerning Network for Repetition-Aware Recommendation. In *Proceedings of the ACM Web Conference 2023 (WWW '23), April 30–May 04, 2023, Austin, TX, USA*. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3543507.3583459

1 INTRODUCTION

Recent years have witnessed the fast development of recommendation system in e-commerce scenario [18, 30, 45, 47]. Based on the user-item interactions, the recommendation system aims to recommend a few items that the user un-interacted previously [1, 4, 26, 27]. In this scenario, the repeated interaction is a common phenomenon, as users usually purchase items of daily use repeatedly, such as shampoo or tissues, etc. With the accumulation of interaction data, the repeated interactions are gradually accounting for a large portion in e-commerce. The statistic from a popular commercial

*Corresponding author

WWW '23, April 30–May 04, 2023, Austin, TX, USA

ACM ISBN 978-1-4503-9416-1/23/04…$15.00
https://doi.org/10.1145/3543507.3583459

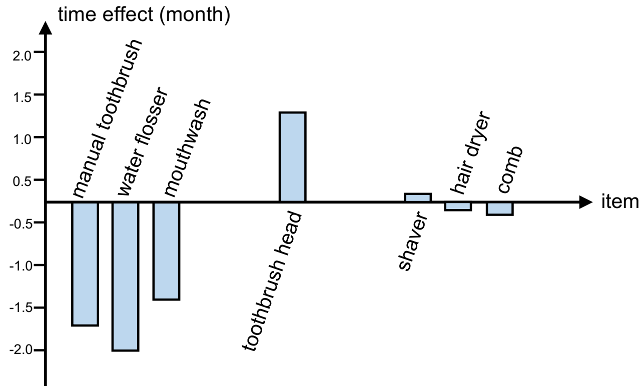

Figure 1: An example to illustrate the influence of other items on the re-consumption of an electric toothbrush. The x-axis denotes the different items, and the positive and negative values in the y-axis denote the average promoted or delayed time to a re-consumption of the electric toothbrush when the corresponding item is consumed.

platform Meituan [1] shows that more than 56% of click data are repeated ones.

The huge amount of repeated interaction data and its potential profit drive various methods produced to infer users' repeated interactions. Among these models, the traditional sequential recommendation models account for a widely applied approach [17, 20, 32]. Recently, some works also further concerned repetition-aware signals [6, 35, 37], and achieved promising results. Though effective, to the best of our knowledge, none of these works have given a qualitative analysis over the item relevance, while we argue the item relevance is a significant factor to determine the next repeated interaction.

To explain this, we give a real example collected from the online data of Meituan, as shown in Figure 1. We found that after the user purchased an electric toothbrush, several of its succeeding interacted items will affect its re-consumption: The re-consumption time of the electric toothbrush is advanced if the user bought its accessory toothbrush head; while the time is obviously delayed if its substitutes are taken (water flosser, mouthwash, and manual toothbrush); in addition, there is no significant impact after interacting a hair dryer or a shaver.

Considering the different incentives that items bring, given a target item, mixing all information together to determine its repetition destiny seems not to be a wise choice (i.e., irrelevant items may become noise misleading the decision, and items having stimulative or inhibitive incentives may require different models to learn their unique properties). It is necessary to discern this relevance and model them separately for a better repetition-aware recommendation.

Although it is appealing in theory, it is non-trivial to realize an effective discernment strategy without any supervision signals. Recently, some attention-based approaches are proposed to automatically assign different weights to each item for relevance estimation. However, the method of assigning weights in $[0, 1]$ still fails to discern various types distinctly, while the accumulation of data with noisy and inconsistent relevance may still complicate the learning process. Besides, the interpretability is also limited. How to automatically discern item relevance for a correct repetition-aware recommendation, is still a crucial and challenging problem.

In light of these opportunities and challenges mentioned above, in this paper, we introduce a novel **C**ommunication M**A**RL-based **R**elevance **D**iscerning Network (**CARD** for short) to automatically discern item relevance for a better repetition-aware recommendation. The main advantage of our **CARD** is that it formalizes the relevance discerning problem into a communication selection task in MARL. Specifically, **CARD** treats each unique interacted item as an agent, and totally considers three different communication types among the designed agents, which are stimulative, inhibitive, and noisy type respectively. The defined types correspond to a stimulative effect, inhibitive effect, and none effect. For each time step, **CARD** utilizes a Gumbel-enhanced classifier to distinguish the communication types among agents, and an attention-based Reactive Point Process is further designed to communicate the well-discerned stimulative and inhibitive incentives separately. Based on the learned communication protocol, **CARD** can well coordinate the behavior of multiple agents for efficient model optimization. We evaluate the effectiveness of the proposed model on both online and offline datasets. For comparison, we consider several well-known sequential and repetition-aware recommenders. The empirical results show that our model can significantly outperform all the baselines in terms of all the evaluation metrics. In total the contributions of our work are as follows:

- We formalize the item relevance discerning problem into a communication selection process in MARL. To the best of our knowledge, this is the first time that Communicative MARL has been explicitly discussed and utilized in the recommendation scenario.
- We define three different communication types to illustrate the interactions among agents, and a Gumbel-enhanced classifier is utilized to automatically discern them. After this, an attention-based Reactive Point Process is further designed to offer an effective communication transmission among agents for each repetition decision.
- Empirical results on both online and offline datasets show that our model can consistently outperform state-of-the-art baselines under different metrics, including sequential and repetition-aware baselines.

2 RELATED WORK

In this section, we provide a brief overview of the related work from three perspectives, including sequential recommendation, repetition-aware recommendation, and communication in multi-agent reinforcement learning respectively.

Sequential Recommendation. Sequential recommendation aims to predict users' future behaviors given their historical interaction data. During the past decades, sequential models gradually evolved from modeling low-order sequential dependencies [38, 46] to high-order ones [11, 24, 29]. Recently, due to the effectiveness of the attention mechanism, several models also utilized this strategy of assigning weights on different items to reveal their significance with the target item and achieved the state-of-the-art performance [20, 25, 40, 54].

[1] Meituan is one of the largest platforms providing local consumer products and retail services in the world.

Due to its nature similarity to repetition-aware recommendation task, these models can seamlessly be transformed for repetition-aware recommendation task with a slight adjustment.

Repetition-aware Recommendation. The repeated user-item interactions now are widely observed in many fields, such as web revisitation [2, 21, 48], repeated queries [41–43], information refinding [9, 14, 22, 28], online recommendation [5, 36, 44], etc.

Previous works usually focused on detecting informative properties that may be a benefit to the re-consumption performance. For example, Anderson [5] studied the patterns by which a user consumed the same item repeatedly over time, and developed a hybrid model to predict users' repeated choices based on a combination of recency and quality. Benson et al.[7] studied several factors that may affect users' repeat behaviors and found the time factor turns out to be the most influential one. Similar conclusions are also found in [44]. Recently, deep models are also utilized to model the repetition-aware signals for an effective recommendation. For example, Ren et al . [37] proposed a model with an encoder-decoder architecture for session-based recommendation. In their work, a repeat-explore mechanism was incorporated into RNN to capture the repeat-aware recommendation intents. Rappaz et al. [35] proposed a self-attentive model and added time interval embeddings to learn temporal dependencies for repetition-aware recommendation.

As we can see, though repetition-aware recommendation has been utilized in various domains, none of these works have considered the influence of different item relevance bring, thus the performance might be limited.

Communication in Multi-agent Reinforcement Learning. Communication is one of the core components for learning coordinated behavior in multi-agent systems, which can significantly improve the flexibility and adaptiveness of a multi-agent system. Unrestricted restricted [3, 19, 31, 49, 50] and restricted communication [12, 23, 33, 39] strategy are two main variants. The unrestricted communication strategy stands for a fully connected structure over agents, which is often used in early works of communication MARL. For example, Sukhbaatar et al.[39] and Peng et al.[33] learned a communication protocol that connected all agents together. Das et al.[12] and Kim et al.[23] learned meaningful messages while using a broadcast way to share messages. However, the full-connected structure requires all-to-all communication among the agents, which can cause significant communication overhead and latency. Different from the unrestricted strategy, the restricted one allows each agent to communicate with a limited number of agents due to partial observability. For example, Zhang et al.[49] introduced a Variance Based Control to reduce the information transferred between the agents. Jiang et al. [19] utilized a graph to describe communication among agents, and each agent was only allowed to communicate with their neighbors.

In our work, we formalize the item relevance discerning problem into a communication selection process in MARL and learn an appropriate communication protocol for better optimization.

3 PRELIMINARY

Notations. Let $\mathcal{U}$ denote a set of users and $\mathcal{V}$ denote a set of items. For each user $u \in \mathcal{U}$, we use $v^u_{1:n} = \{(v^u_1, t^u_1), (v^u_2, t^u_2), \dots, (v^u_n, t^u_n)\}$ to denote the interaction sequence for user u, where $v^u_k \in \mathcal{V}$ represents the k-th item that u has interacted , t^u_k corresponds its interaction time, and n is the sequence length. For simplicity, We drop the superscript of u in the notations for ease of reading.

Task Definition. Given an interaction sequence, we are interested in whether a repeated interaction will be triggered in his/her next visit, and which item will be re-interacted.

Communication in Multi-agent Reinforcement Learning. Communication is an effective mechanism for coordinating the behaviors of multiple agents. Here we first give a brief introduction about the communication in MARL [8], and the framework is shown in Fig. 2.

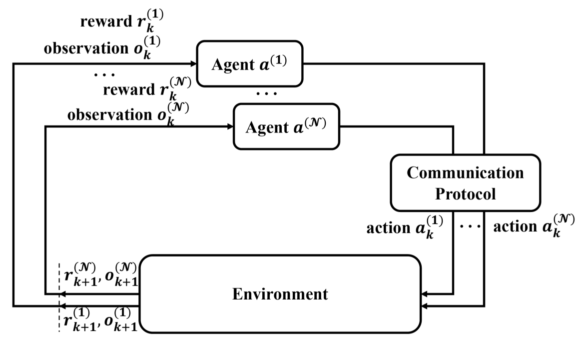

Figure 2: The overall architecture of Communication in MARL.

The communicative MARL can be described as a stochastic game G, represented as a tuple $G = \{\mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{T}, \mathcal{M}, r, \gamma\}$, where $\mathcal{N}$ represents the agent count; $\mathcal{S}$ is the set of states and $s_k \in \mathcal{S}$ represents the k-th state; $\mathcal{A} = \{A^{(1)}, A^{(2)}, \cdots, A^{(\mathcal{N})}\}$ is the collection of action sets, with $a^{(i)}_k \in A^{(i)}$ being i-th agent's action at k-th step; $\mathcal{O} = \{O^{(1)}, O^{(2)}, \cdots, O^{(\mathcal{N})}\}$ is the set of observations; $\mathcal{T}$ is the state transition function: $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$; $\mathcal{M}$ represents the space of messages and γ is a discount factor. For the i-th agent at the k-th step, it receives a private observation $o^{(i)}_k \in O^{(i)}$ and message $m^{(i)}_k \in \mathcal{M}$ to output an action $a^{(i)}_k$, and obtains a reward $r^{(i)}_k$.

4 METHODOLOGY

In this section, we introduce the proposed **CARD** in detail, and the overall architecture of **CARD** is presented Figure 3 . In the following, we start with a communicative MARL for our task, then present our communication learning strategy for model optimization.

4.1 The Communicative MARL framework

First, we use Communicative MARL to frame our task. In a Communicative MARL, each agent is responsible for a unique item in the interaction sequence. By communicating with other agents, it aims to interact with the environment at the discrete time.

Specifically, given an interaction sequence $v_{1:n}$, we assign an agent to respond each unique interacted item in $v_{1:n}$, by this we can obtain an agent sequence, denoted as $e_{1:\mathcal{N}} = \{e_1, e_2, \dots, e_{\mathcal{N}}\}$, where the agent count $\mathcal{N}$ is equal to the count of unique items in $v_{1:n}$. For simplicity, we use a lookup table to map them, denoted as $e_i = map(v_k)$, indicating the i-th unique item in $v_{1:n}$ was interacted at

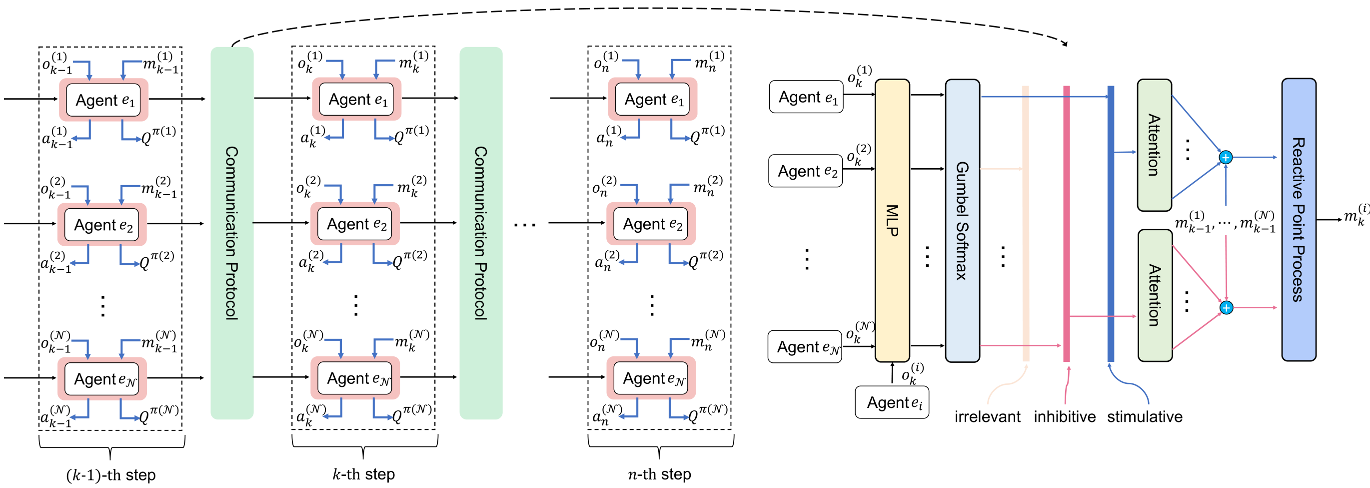

Figure 3: The overall architecture of Communicative MARL-based Relevance Discerning Network (CARD for short). CARD formalizes the item relevance discerning problem into a communication learning process. Based on the defined three communication types, CARD discerns them according to a Gumbel-enhanced classifier, and utilizes an attention-based Reactive Point Process to aggregate them separately for effective transmission.

k-th step. Given the interacted item v_k, the observation embedding of the corresponding agent e_i can be written as:

$$\mathbf{o}_k^{(i)} = GRU[\mathbf{o}_{k-1}^{(i)}, I(a_{k-1}^{(i)} = 1)\mathbf{v}_k] \tag{1}$$

where $\mathbf{v}_k$ is the embedding of the k-th item, $\mathbf{o}_{k-1}^{(i)}$ is the previous observation embedding and $I(\cdot)$ represents the identity function. Note that as agent e_i was activated until its corresponding item has interacted, thus we set $\mathbf{o}_{k-1}^{(i)}$ is a zero vector when the agent was not active.

After this, we utilize an action-value network $Q^{\pi(i)}(o_k^{(i)}, a_k^{(i)}, m_k^{(i)})$ to make actions through a Multi-Layer Perception (MLP), where $a_k^{(i)} \in$ {repeating =1,non-repeating=0 }, and $m_k^{(i)}$ represents messages other agents transmitted to agent e_i at the k-th step, which will be discussed in the later section. In our work, we use 1 to represent the item that is re-interacted, and 0 otherwise. After each action, the agent e_i receives a numerical intermediate reward $r_k^{(i)}$, and we set $r_k^{(i)}$=1 if the item that agent e_i corresponded was re-interacted successfully in the k-th step, else $r_k^{(i)}$=0.

We then collaborate on agents' actions for better collaboration. Given the joint action-values, the loss function is written as:

$$\mathcal{L}(\Theta) = \sum_{u\in\mathcal{U}} \sum_{k} \sum_{e_i \in e_{1:\mathcal{N}_{1:n}}} \Big(r_k^{(i)} + \gamma \max_{a_{k+1}^{(i)}} Q^{\pi(i)}(o_{k+1}^{(i)}, a_{k+1}^{(i)}, m_{k+1}^{(i)}) - Q^{\pi(i)}(o_k^{(i)}, a_k^{(i)}, m_k^{(i)}); \Theta\Big)^2 \tag{2}$$

where Θ represents all parameters need to learn.

4.2 Learning Communication Protocol for MARL

A key point for Communicative MARL is to model and learn an effective communication protocol. With effective communication, agents can obtain a better understanding of the environment for an effective repetition-aware recommendation. Specifically, **CARD** utilizes a Gumbel-enhanced classifier for communication selection, and an attention-based Reactive Point Process to manage the communication transmission process among agents. In the following, we will give a detailed analysis of such a design.

4.2.1 Gumbel-enhanced Communication Classifier. Given agent e_i and its neighbor e_j, **CARD** can utilize a simple classifier to distinguish their communication type at the k-th time step, which is written as follows:

$$P(m_{i,j}|e_i, e_j) = \frac{(\mathbf{o}_k^{(i)} \oplus \mathbf{o}_k^{(j)})\mathbf{w}_{m_{i,j}}}{\sum_{m_{i,j}\in M}(\mathbf{o}_k^{(i)} \oplus \mathbf{o}_k^{(j)})\mathbf{w}_{m_{i,j}}} \tag{3}$$

where $\oplus$ is a concatenation operator, $\mathbf{M} = \{stimulative = 0, inhibitive = 1, noisy = 2\}$ represents all three communication types **CARD** defined, where $m_{i,j} \in \mathbf{M}$, and $\mathbf{w}_{m_{i,j}}$ is the corresponding parameters for each type. Note that at the k-step, neighbors of agent e_i contain all its succeeding activated agents, where $j > i$ and $j < k$.

However, this traditional hard-coding mechanism is not differentiable and prevents the model from being trained well via backpropagation. To address this issue, inspired by [34], we integrate a Gumbel Softmax into the classifier as a differentiable surrogate to support model learning over the discrete output. Specifically, the enhanced classifier can be written as:

$$P(m_{i,j}|e_i, e_j) = \frac{\exp\big(\big(\log(MLP(\mathbf{o}_k^{(i)}, \mathbf{o}_k^{(j)})) + \epsilon_{m_{i,j}}\big)/\tau\big)}{\sum\limits_{m_{i,j}\in\mathbf{M}} \exp\big(\big(\log(MLP(\mathbf{o}_k^{(i)}, \mathbf{o}_k^{(j)})) + \epsilon_{m_{i,j}}\big)/\tau\big)}$$

where $\epsilon_{m_{i,j}}$ represents a noise sampled from a Gumbel distribution, and the temperature parameter τ controls its sharpness. When τ is small,$P(m_{i,j}|e_i, e_j)$ produces a multi-modal distribution. On the contrary, it approximates a one-hot vector.

4.2.2 Attention-based Reactive Point Process. After the communication discerning process, for each agent e_i, its communicated agents are further divided into 3 sub-groups: a stimulative subgroup $\mathcal{G}_{st}^i$ having stimulative incentives to agent e_i, an inhibitive

Table 1: Statistics of datasets for experiments (a.v.l=average sequence length).

Datset	#interactions	#users	#items	#a.v.l	repeat ratio
Meituan	7,123,343	100,000	1,592,172	74	0.41
Alibaba	576,440	10,618	130,862	54	0.35

sub-group $\mathcal{G}_{in}^i$ preserving agents with inhibitive incentives, and a noisy sub-group $\mathcal{G}_{no}^i$ keeping irrelevant ones. Afterward, it becomes distinct to model them respectively. Specifically, we use the following function to aggregate all useful information that other agents transmitted at the k-th step:

$$\alpha_{k,st}^{i,j} = \frac{e^{(cos(\mathbf{o}_k^{(i)},\mathbf{o}_k^{(j)}))}}{\sum\limits_{j \in \mathcal{G}_{st}^i} e^{(cos(\mathbf{o}_k^{(i)},\mathbf{o}_k^{(j)}))}}; \alpha_{k,in}^{i,j} = \frac{e^{(cos(\mathbf{o}_k^{(i)},\mathbf{o}_k^{(j)}))}}{\sum\limits_{j \in \mathcal{G}_{in}^i} e^{(cos(\mathbf{o}_k^{(i)},\mathbf{o}_k^{(j)}))}}$$

where $cos(\cdot)$ represents a cosine function. Note that we only consider the stimulative and inhibitive communication type in **CARD** , and remove agents with the noisy type to keep the effectiveness of the fused information. After this, given an agent e_i and one of its neighbors e_j, we then use the following function to model the message that agent e_j transmitted to e_i:

$$\mathbf{m}_k^{i,j} = [I(m_{i,j} = 0)\alpha_{k,st}^{i,j} + I(m_{i,j} = 1)\alpha_{k,in}^{i,j}] \times \mathcal{J}(t_{k+1}^u - t_k^u) \times \mathbf{o}_k^{(j)} + \mathbf{m}_{k-1}^{i,j}$$

where $\mathcal{J}(t_{k+1}^u - t_k^u) = \beta \log(1 + \exp(t_{k+1}^u - t_k^u)/\beta)$ is a kernel function [55] that models the decaying time effect between two interactions.

Based on the learned transmitted message, we then use a Reactive Point Process [15] to fuse the discerned messages linearly for the final message representation $\mathbf{m}_k^i$ that agent e_i received from e_j:

$$\mathbf{m}_k^i = \underbrace{\boldsymbol{\lambda}^i}_{base} + \underbrace{\sum_{j \in \mathcal{G}_{st}^i} \mathbf{m}_k^{i,j}}_{stimulative} - \underbrace{\sum_{j \in \mathcal{G}_{in}^i} \mathbf{m}_k^{i,j}}_{inhibitive} \tag{4}$$

where $\boldsymbol{\lambda}^i$ is a bias representing its single self-exciting effect. according to such a design, **CARD** can fuse stimulative and inhibitive incentives separately, while discarding useless communication. Based on the well-learned communication protocol, we believe our **CARD** can well handle the collaboration over agents to achieve the maximum expected total reward for all agents.

4.3 Learning and Recommendation

In order to learn the parameters of **CARD** , we employ the Double Q-learning algorithm to train it according to the Eq.2. However, training **CARD** is impossible as we need to maintain the same equivalent agents with items, which is extremely huge. Inspired by [19], we then follow the parameter-sharing strategy, where each agent independently learns an action-value network with fully shared parameters among all agents.

With the learned **CARD** , given an interaction sequence $v_{1:n}$, we obtain the final observations and messages of all agents according to Eq. 1 and Eq. 4. Note that we have known the real actions of each agent given $v_{1:n}$, different from the training stage, we can directly assign corrective actions for each agent to produce perfect observation embeddings. We then utilize this trick to obtain better observations for the following recommendation task.

Based on the tricky observations and the learned messages, each agent gives its next action according to the shared action-value network, and the recommendation strategy is summarized as follows: If all agents choose none-repeating actions, then **CARD** judges the user will not perform a repeated interaction; Otherwise, **CARD** keeps agents having repeating actions, and ranks them according to their action values. We then select the item that the Top-1 agent mapped as the re-interacted one.

5 EXPERIMENT

In this section, we evaluate **CARD** by comparing it with both the sequential and repetition-aware recommenders. We begin by introducing the experimental setup and analyze the experimental results.

5.1 Experimental Setup

Dataset. We conduct our experiments on two real-world datasets:

- **Meituan**: We collected the click behaviors of 100k users from search advertising system in Meituan, ranging from Jul.2022 to Sep. 2022.
- **Alibaba**[2]: A dataset released by Alibaba e-commerce platform. It covers the shopping behavior in 22 days of millions of users. we randomly sample 1% of users by considering the diversity of datasets.

For all datasets, we remove users and items with fewer than 3 related actions. The statistics of the two datasets are shown in Table 1. We then follow the leave-one-out evaluation strategy to split each interaction sequence into three parts: the last item of each sequence for testing; the next-to-last item for validation; and the rest for training.

Baselines.To evaluate the effectiveness of our approach, We compare **CARD** against two types of baselines, which are sequential recommenders and repetition-aware recommenders respectively. The sequential recommenders include:

(1) FPMC [38]: FPMC is a shallow model that combines matrix factorization and factorized first-order Markov chains for sequential recommendation.
(2) GRU4Rec [11]: GRU4Rec is a session-based recommendation, which utilizes GRU unit to capture users' long sequential behaviors for recommendation.
(3) SASRec [20]: SASRec is a self-attention based sequential recommendation model, which uses the multi-head attention mechanism to recommend the next item.
(4) TiSASRec [25]: TiSASRec is a time interval aware self-attention based sequential recommendation, which models both the absolute positions of items as well as the time intervals between them in a sequence.

For repetition-aware recommenders, we consider the following baselines:

(1) ReCANet [6]: ReCANet proposes a framework with LSTM layers to explicitly model the repeat consumption behavior of users.

[2]https://tianchi.aliyun.com/dataset/dataDetail?dataId=56

Table 2: Performance comparison between baselines and CARD . The best performance of each column is highlighted in boldface. Symbol $*$ denotes the best baseline. Symbol ▲ denotes the relative improvement of our results against the best baseline, which are consistently significant at 0.05 level.

Task	Dataset	Metric	Sequential Models				Repeat-aware Models					▲%
			FPMC	GRU4Rec	SASRec	TiSASRec	GRU4Rec$_H$	ReCANet	LiveRec	SASRec$_H$	CARD	
Task 1	Meituan	AUC	0.5300	0.5465	0.5532	0.5545	0.5624	0.5657	0.5792	0.5811*	**0.5967**	2.68
		LogLoss	0.9365	0.8731	0.8526	0.8238	0.7853	0.7820	0.7558	0.7316*	**0.6896**	5.74
		Accuracy	0.5057	0.5109	0.5204	0.5345	0.5411	0.5432	0.5643	0.5764*	**0.6139**	6.50
	Alibaba	AUC	0.5265	0.5375	0.5510	0.5670	0.5701	0.5869	0.6121	0.6233*	**0.6347**	1.83
		LogLoss	8.5815	2.0701	1.8924	1.4146	1.2323	1.0896	0.9452	0.7579*	**0.6698**	11.62
		Accuracy	0.5283	0.5623	0.5774	0.5976	0.6021	0.6128	0.6339	0.6575*	**0.6743**	2.55
Task 2	Meituan	AUC	0.6082	0.6427	0.6786	0.7335	0.7585	0.7614	0.8136	0.8376*	**0.8925**	6.55
		LogLoss	0.2784	0.2721	0.2622	0.2576	0.2501	0.2433	0.2376	0.2299*	**0.2106**	8.39
		Accuracy	0.1888	0.2581	0.3631	0.4256	0.4947	0.5257	0.5526	0.5712*	**0.6123**	7.18
	Alibaba	AUC	0.8257	0.8529	0.8641	0.8858	0.8995	0.9052	0.9165	0.9315*	**0.9510**	1.99
		LogLoss	0.3159	0.3102	0.3067	0.2846	0.2752	0.2641	0.2574	0.2214*	**0.2048**	7.50
		Accuracy	0.4987	0.5384	0.6396	0.6745	0.7114	0.7320	0.7616	0.7975*	**0.8487**	6.42

(2) GRU4Rec$_H$: GRU4Rec$_H$ is an extension of GRU4Rec which incorporates a neural Hawkes process model [13] to model both re-consumption signals and time intervals.
(3) LiveRec [35]: LiveRec is a live-steaming recommender incorporating both the recurring consumption patterns and time intervals for recommendation.
(4) SASRec$_H$: Similar to GRU4Rec$_H$, we extend SASRec with a transformer Hawkes process [55].

Evaluation Metric. We employ the commonly used AUC(Area Under ROC), LogLoss (cross-entropy) [16] and Accuracy to assess the performance of the two mentioned tasks. We perform significant tests using the paired t-test. Differences are considered statistically significant when the p-value is lower than 0.05.

Parameter Settings. For a fair comparison, the batch size is fixed to 512, and the latent dimension for all models is 32. The parameters are normally initialized with 0 mean. All normal initializers have 0.01 standard deviation. For LiveRec[3] and ReCANet [4], we use the source code provided by their authors. For other methods, we implement them by RecBole [51]. We optimize them according to the validation sets.

For our model, we implement it based on the PyTorch framework. The discount factor γ is set to 0.99, and β in the kernel function is set to 2. In the sampling stage, we preserve 5 episodes in our replay buffer for each sequence.

5.2 Performance Comparison

In this section, we compare the performance of our model with the baselines. We totally consider the following two tasks:

- **Task 1**: A binary classification task considering whether a repeated interacted interaction will be triggered at the next visit time.
- **Task 2**: A multi-classification task concerning once a repetition interaction is confirmed, which item will be most likely re-interacted?

[3]https://github.com/JRappaz/liverec
[4]https://github.com/mzhariann/recanet

The overall performance of our proposed **CARD** and the baselines are reported in Table 2. We have the following observations:

For sequential recommenders, FPMC obtains the worst performance by capturing only low-level dependencies in sequences. Compared with FPMC, we found the high-order dependency is an important factor for repetition-aware recommendation. This observation demonstrates the necessity of introducing more item-level relevance for a better repetition-aware recommendation. Compared with GRU4Rec, SASRec obtains a better performance by utilizing the attention mechanism of assigning weights on items to discern their significance. This coincides with our assumption that it is necessary to make a detailed analysis of the item's relevance for better performance. By considering the time intervals among items, TiSASRec achieves the best performance among the traditional sequential recommenders.

We found that repetition-aware recommenders perform better than sequential recommenders in most cases. It demonstrates repetition recommendation problem has its own specific characteristics, where we need to explore the informative repetition-aware signals for a better result. Compared with the ReCANet using a LSTM layer to capture the re-consumption patterns, GRU4Rec$_H$ utilizes a Hawkes Process to model the self-exciting of each item and obtains a better performance. LiveRec and SARRec$_H$ further utilize a transformer to model the significance among items and perform better than GRU4Rec$_H$.

Finally, our proposed approach **CARD** achieves the best performance among all the methods on two datasets. Although the baseline SARRec$_H$ unitized the temporal point process and the attention mechanism to consider the influence that time intervals and item dependencies bring , it fails to give a distinct discernment for item relevance. The major contribution of **CARD** is that we formalize the item relevance discerning problem as a communicative MARL framework. **CARD** treats each unique interacted item as an agent and defines three different communication types over agents. By utilizing a Gumbel-enhanced classifier to distinguish the communication types among agents, **CARD** uses an attention-based Reactive Point Process to aggregate the well-classified incentives. With such a meaningful design, **CARD** outperforms the state-of-the-art recommendation methods. Take Meituan dataset as

Table 3: Performance comparison of CARD and its two variants over two datasets. The Best performance is in bold font.

Task	Dataset	Metric	CARD [1]	CARD [2]	CARD
Task 1	Meituan	AUC	0.5733	0.5816	**0.5967**
		LogLoss	0.7612	0.7335	**0.6896**
		Accuracy	0.5622	0.5892	**0.6139**
	Alibaba	AUC	0.6120	0.6221	**0.6347**
		LogLoss	0.8843	0.7369	**0.6698**
		Accuracy	0.6458	0.6625	**0.6743**
Task 2	Meituan	AUC	0.8082	0.8359	**0.8925**
		LogLoss	0.2301	0.2256	**0.2106**
		Accuracy	0.5647	0.5712	**0.6123**
	Alibaba	AUC	0.9187	0.9237	**0.9510**
		LogLoss	0.2430	0.2351	**0.2048**
		Accuracy	0.7683	0.7912	**0.8487**

an example, when compared with the best baseline (i.e., SRSRec$_H$) in Task 2, the performance improvement of **CARD** in terms of relative value is around 1.99% , 7.5%, and 6.42% on AUC, LogLoss, and Accuracy respectively.

5.3 Ablation Study

CARD learns an effective communication protocol for a better recommendation. To achieve this, **CARD** introduces a communication discernment strategy to atomically discern the communication types among agents, and a transmission strategy to fuse the classified incentives for recommendation. In this section, we conducted experiments to compare different implementations of the two informative approaches used in **CARD** to give a deep understanding of the model.

5.3.1 Effectiveness of Discernment Strategy. In **CARD** , we consider three different communication types over agents. To verify the effectiveness of such a design, we compare it with its two degraded versions:

- **CARD** ^{1}: we only define one communication type among agents, by this, each agent in **CARD** ^{1} degrades to a Hawkes Process enhanced unit, which is quite similar to GRU4Rec$_H$;
- **CARD** ^{2}: **CARD** ^{2} considers two communication types, which are relevant and irrelevant respectively. Compared with **CARD** , **CARD** ^{2} considers a denoising problem of removing irrelevant communications but mixing stimulative and inhibitive ones for optimization.

Table 3 shows the performance **CARD** and its two variants on two datasets.

We can see that **CARD** ^{1} obtains the worst performance, it demonstrates the necessity of communication discernment for the repetition-aware recommendation. By removing the irrelevant types, both **CARD** ^{2} and **CARD** perform better than **CARD** ^{1}. Furthermore, compared with **CARD** ^{2}, **CARD** gives a more fine distinction between the stimulative and inhibitive incentives. By fusing them according to Eq. 4 separately, **CARD** is more able to capture their unique properties and obtain the best performance.

5.3.2 Effectiveness of Transmission Strategy. Recall **CARD** utilizes an attention-based Reactive Point Process to direct the communication transmissions among agents according to Eq. 4. In this section,

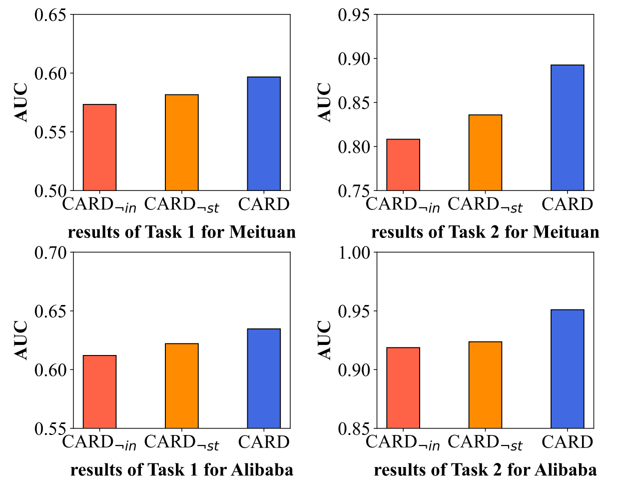

Figure 4: Comparisons between CARD and its two variant models CARD $_{\neg st}$ and CARD $_{\neg in}$ in term of AUC on two tasks.

we design various transmission strategies to analyze the effect of such design. Specifically, we mask the stimulative factor in Eq 4, by this agents having stimulative incentives are ignored, and we name the new model as **CARD** $_{\neg st}$. Similarly, we mask the inhibitive part and name the model as **CARD** $_{\neg in}$. Figure 4 shows the performance between **CARD** and its two sub-models in terms of AUC on two datasets.

An interesting observation is that performance **CARD** $_{\neg st}$ is slight better than **CARD** $_{\neg in}$ on two datasets. The reason might be that items with inhibitive type hold the majority in the interaction sequence. By capturing such influential factors, **CARD** $_{\neg st}$ performs better than **CARD** $_{\neg in}$. Compared with its two sub-models, **CARD** obtains the best performance. It demonstrates that both stimulative and inhibitive communication can contribute to performance in their own way. By considering the effects of different communication types separately, **CARD** can well model evolutions of agents for a correct repetition prediction. Take Meituan dataset as an example, when compared with the **CARD** $_{\neg st}$ on Task 2 , the performance improvement of **CARD** in terms of absolute value is around 2.1% on Accuracy.

5.4 Analysis on Sequence Length

To further investigate the performance of different methods, we split the users into three groups(i.e. short, medium, and long) according the sequence length. By this, we aim to check the influence of sequence length on the recommendation performance. We conduct the comparisons on different groups on Task 2. Take Meituan dataset as an example, a user is classified into the short group if the sequence length is less than 25, and long if it is larger than 50. The remaining users are taken as the medium. In this way, the proportions of short, medium, and long are 19%, 55%, and 26% respectively. For simplicity, we compare our model with SASRec$_H$, and report the performance on Meituan dataset, similar results are also found in Alibaba dataset. The results are shown in Figure 5.

From the results, we can see that **CARD** consistently achieves significant performance gain against SASRec$_H$. This further validates

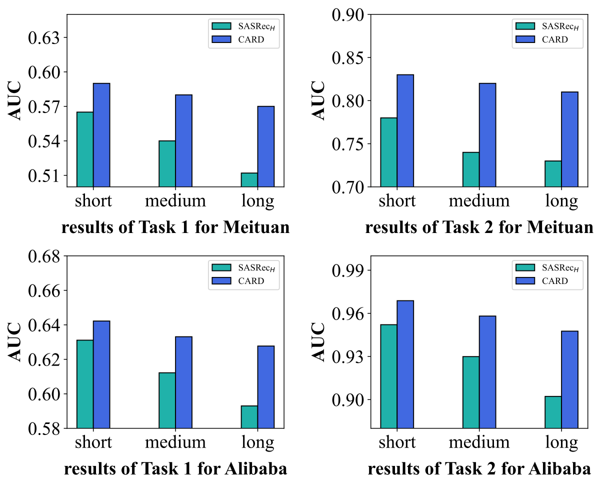

Figure 5: Performance comparison of CARD and SASRec$_H$ on two datasets over different user groups.

the effectiveness of **CARD** for a better repetition-aware recommendation. In addition, we found when the sequence length grows, the performance of both **CARD** and SASRec$_H$ decreases. This is consistent with the expectation as a long interaction sequence indicates more repeat candidates, which will complicate the recommendation process. However, we also found the performance gain between SARRec$_H$ and **CARD** increases as the sequence grows. It implies that **CARD** is more able to discern the item-level relevance according to a learned communication protocol, thus obtaining a better result.

5.5 Online A/B Testing

CARD is designed for the repetition-aware recommendation, we also want to check whether it can bring benefit to other online models. To test the capacity of **CARD** , we select three famous online models deployed in Metituan search advertising system , which are DIN[53], DIEN[52], and BST[10] respectively. The integration process between **CARD** and selected online models is designed as follows: when **CARD** detects a repeat interaction, the action-value of each agent obtained is treated as the repetition-aware feature, which is further fed into the selected models for their corresponding tasks. We conducted online A/B testing on the system from 2022-09-20 to 2022-09-26 for online serving. CTR and eCPM are used for evaluation metrics. The experiment results are shown in Table 4.

We can see that by integrating **CARD** with online models, the performance of all models improved in terms of CTR and eCPM. It

Table 4: Performance of DIN, DIEN and BST when combining with CARD in an Online A/B Testing.

Models	Online metrics	
	CTR Gain	eCPM Gain
CARD + DIN	1.12%	0.75%
CARD + DIEN	1.13%	0.82%
CARD + BST	1.52%	1.21%

Table 5: The distributions of communication types over two datasets.

Dataset	Communication types		
	stimulative	inhibitive	noisy
Meituan	24.21%	32.78%	42.01%
Alibaba	11.56%	27.30%	61.14%

demonstrates that **CARD** is not only competent for a repetition-aware recommendation, it can also provide effective features for various online tasks.

5.6 Further Analysis

In this section, we analyze the significance of different communication types for a correct recommendation over each dataset. Specifically, for each interaction sequence that **CARD** recommends correctly in the testing set, we statistic the communication type of each item in the interaction sequence. Based on these types, we calculate their percentages. The percentage distributions on two datasets are shown in Table 5.

An interesting observation is that most items are irrelevant, and the result is quite consistent on both Meituan and Alibaba datasets. It demonstrates the necessity of denoising in this specific scenario. Furthermore, For the rest types, we found the distribution of the inhibitive type is larger than the stimulative one. This coincides with our previous finds in Ablation Study, where inhibitive communication is more important in determining a correct repetition-aware recommendation due to its large count. Overall, **CARD** is able to discard the noisy communication, after that, an attention-based Reactive Point Process is further utilized to aggregate the discerned stimulative and inhibitive communication. According to such an appropriate design, **CARD** gives a high-quality communication protocol for effective collaborations over agents and obtains the best performance.

6 CONCLUSION

In this paper, we address an item-level relevance discerning problem in repetition-aware recommendation scenario. We formalize this task into a communication selection process in MARL, and a novel **C**ommunicative M**A**RL-based **R**elevance **D**iscerning Network (**CARD** for short) is designed to learn an effective communication protocol for a recommendation. Based on the defined three different types of agents, **CARD** utilizes a Gumbel-enhanced classifier to discern these types, and an attention-based Reactive Point Process is further designed to transmit the well-discerned incentives for a better recommendation. Experiments on both online and offline datasets demonstrate the effectiveness of our proposed model.

To the best of our knowledge, it is the first time learning constrained communication in MARL to discern item-level relevance in a repetition-aware recommendation scenario. Currently, our focus lies in the utilization of item-level relevance in the MARL framework. In future work, we will consider to fuse more context-aware information to analyze the relevance among heterogeneous entities for further improvement.

ACKNOWLEDGMENTS

This research work was supported by fundamental Research for the National Natural Science Foundation of China (No.62032003). We would like to thank the anonymous reviewers for their valuable comments.

REFERENCES

[1] Panagiotis Adamopoulos and Alexander Tuzhilin. 2014. On Unexpectedness in Recommender Systems: Or How to Better Expect the Unexpected. *TIST* 5, 4 (2014), 54:1–54:32.

[2] Eytan Adar, Jaime Teevan, and Susan T. Dumais. 2008. Large scale analysis of web revisitation patterns. In *CHI 2008*. 1197–1206.

[3] Akshat Agarwal, Sumit Kumar, Katia P. Sycara, and Michael Lewis. 2020. Learning Transferable Cooperative Behavior in Multi-Agent Teams. In *AAMAS 2020*. 1741–1743.

[4] Noor Aldeen Alawad, Aris Anagnostopoulos, Stefano Leonardi, Ida Mele, and Fabrizio Silvestri. 2016. In *SIGIR 2016*. 913–916.

[5] Ashton Anderson, Ravi Kumar, Andrew Tomkins, and Sergei Vassilvitskii. 2014. The Dynamics of Repeat Consumption. In *WWW 2014*. 419–430.

[6] Mozhdeh Ariannezhad, Sami Jullien, Ming Li, Min Fang, Sebastian Schelter, and Maarten de Rijke. 2022. ReCANet: A Repeat Consumption-Aware Neural Network for Next Basket Recommendation in Grocery Shopping. In *SIGIR 2022*. 1240–1250.

[7] Austin R. Benson, Ravi Kumar, and Andrew Tomkins. 2016. Modeling User Consumption Sequences. In *WWW 2016*. 519–529.

[8] Craig Boutilier. 1996. Planning, Learning and Coordination in Multiagent Decision Processes. In *TARK 1996*. 195–210.

[9] Jun Chen, Chaokun Wang, and Jianmin Wang. 2015. Will You "Reconsume" the Near Past? Fast Prediction on Short-term Reconsumption Behaviors. In *AAAI 2015*. 23–29.

[10] Qiwei Chen, Huan Zhao, Wei Li, Pipei Huang, and Wenwu Ou. 2019. Behavior Sequence Transformer for E-Commerce Recommendation in Alibaba. In *DLP-KDD 2019*.

[11] Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. In *EMNLP*. 103–111.

[12] Abhishek Das, Théophile Gervet, Joshua Romoff, Dhruv Batra, Devi Parikh, Mike Rabbat, and Joelle Pineau. 2019. TarMAC: Targeted Multi-Agent Communication. In *ICML 2019*, Vol. 97. 1538–1546.

[13] Nan Du, Hanjun Dai, Rakshit Trivedi, Utkarsh Upadhyay, Manuel Gomez-Rodriguez, and Le Song. 2016. Recurrent marked temporal point processes: Embedding event history to vector. In *KDD 2016*. 1555–1564.

[14] David Elsweiler, Mark Baillie, and Ian Ruthven. 2011. What makes re-finding information difficult? a study of email re-finding. In *ECIR 2011*. 568–579.

[15] Şeyda Ertekin, Cynthia Rudin, and Tyler H McCormick. 2015. Reactive point processes: A new approach to predicting power failures in underground electrical systems. *The Annals of Applied Statistics* 9, 1 (2015), 122–144.

[16] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: A Factorization-Machine based Neural Network for CTR Prediction. In *IJCAI 2017*. 1725–1731.

[17] Balazs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2016. Session-based Recommendations with Recurrent Neural Networks. In *ICLR*.

[18] Jin Huang, Wayne Xin Zhao, Hong-Jian Dou, Ji-Rong Wen, and Edward Y. Chang. 2018. Improving Sequential Recommendation with Knowledge-Enhanced Memory Networks. In *SIGIR*. 505–514.

[19] Jiechuan Jiang, Chen Dun, Tiejun Huang, and Zongqing Lu. 2020. Graph Convolutional Reinforcement Learning. In *ICLR 2020*.

[20] Wang-Cheng Kang and Julian J. McAuley. 2018. Self-Attentive Sequential Recommendation. In *ICDM*. 197–206.

[21] Ricardo Kawase, George Papadakis, and Eelco Herder. 2011. Supporting Revisitation with Contextual Suggestions. In *JCDL 2011*. 4 pages.

[22] Ricardo Kawase, George Papadakis, Eelco Herder, and Wolfgang Nejdl. 2010. The impact of bookmarks and annotations on refinding information. In *acm conference on hypertext*. 29–34.

[23] Woojun Kim, Jongeui Park, and Youngchul Sung. 2021. Communication in Multi-Agent Reinforcement Learning: Intention Sharing. In *ICLR 2021*.

[24] Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. 2017. Neural Attentive Session-Based Recommendation. In *CIKM*. 1419–1428.

[25] Jiacheng Li, Yujie Wang, and Julian J. McAuley. 2020. Time Interval Aware Self-Attention for Sequential Recommendation. In *WSDM 2020*. 322–330.

[26] Pan Li, Maofei Que, Zhichao Jiang, Yao Hu, and Alexander Tuzhilin. 2020. PURS: Personalized Unexpected Recommender System for Improving User Satisfaction. In *RecSys 2020*. 279–288.

[27] Pan Li and Alexander Tuzhilin. 2020. Latent Unexpected Recommendations. *TIST* 11, 6 (2020), 70:1–70:25.

[28] Weiyang Lin, Sergio A. Alvarez, and Carolina Ruiz. 2002. Efficient Adaptive-Support Association Rule Mining for Recommender Systems. *Data Min. Knowl. Discov.* 6, 1 (2002), 83–105.

[29] Qiao Liu, Yifu Zeng, Refuoe Mokhosi, and Haibin Zhang. 2018. STAMP: Short-Term Attention/Memory Priority Model for Session-Based Recommendation. In *KDD*. 1831–1839.

[30] Eleanor Loh, Jalaj Khandelwal, Brian Regan, and Duncan A. Little. 2022. Promotheus: An End-to-End Machine Learning Framework for Optimizing Markdown in Online Fashion E-commerce. In *KDD 2022*. 3447–3457.

[31] Aleksandra Malysheva, Tegg Tae Kyong Sung, Chae-Bong Sohn, Daniel Kudenko, and Aleksei Shpilman. 2018. Deep Multi-Agent Reinforcement Learning with Relevance Graphs. *CoRR* abs/1811.12557 (2018).

[32] M. Jeffrey Mei, Cole Zuber, and Yasaman Khazaeni. 2022. A Lightweight Transformer for Next-Item Product Recommendation. In *RecSys 2022*. 546–549.

[33] Peng Peng, Quan Yuan, Ying Wen, Yaodong Yang, Zhenkun Tang, Haitao Long, and Jun Wang. 2017. Multiagent Bidirectionally-Coordinated Nets for Learning to Play StarCraft Combat Games. *CoRR* abs/1703.10069 (2017).

[34] Yuqi Qin, Pengfei Wang, and Chenliang Li. 2021. The World is Binary: Contrastive Learning for Denoising Next Basket Recommendation. In *SIGIR*. 859–868.

[35] Jérémie Rappaz, Julian J. McAuley, and Karl Aberer. 2021. Recommendation on Live-Streaming Platforms: Dynamic Availability and Repeat Consumption. In *RecSys 2021*. 390–399.

[36] Markus Reiter-Haas, Emilia Parada-Cabaleiro, Markus Schedl, Elham Motamedi, Marko Tkalcic, and Elisabeth Lex. 2021. Predicting Music Relistening Behavior Using the ACT-R Framework. In *RecSys 2021*. 702–707.

[37] Pengjie Ren, Zhumin Chen, Jing Li, Zhaochun Ren, Jun Ma, and Maarten de Rijke. 2019. RepeatNet: A Repeat Aware Neural Recommendation Machine for Session-Based Recommendation. In *AAAI 2019*. 4806–4813.

[38] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing Personalized Markov Chains for Next-basket Recommendation. In *WWW*. 811–820.

[39] Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. 2016. Learning Multiagent Communication with Backpropagation. In *NIPS 2016*. 2244–2252.

[40] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. In *CIKM*. 1441–1450.

[41] Jaime Teevan, Eytan Adar, Rosie Jones, and Michael Potts. 2006. History Repeats Itself: Repeat Queries in Yahoo's Logs. In *SIGIR 2006*. 703–704.

[42] Jaime Teevan, Eytan Adar, Rosie Jones, and Michael A. S. Potts. 2007. Information Re-retrieval: Repeat Queries in Yahoo's Logs. In *SIGIR 2007*. 151–158.

[43] Sarah K. Tyler and Jaime Teevan. 2010. Large Scale Query Log Analysis of Re-finding. In *WSDM 2010*. 191–200.

[44] Chenyang Wang, Min Zhang, Weizhi Ma, Yiqun Liu, and Shaoping Ma. 2019. Modeling Item-Specific Temporal Dynamics of Repeat Consumption for Recommender Systems. In *WWW 2019*. 1977–1987.

[45] Pengfei Wang, Hanxiong Chen, Yadong Zhu, Huawei Shen, and Yongfeng Zhang. 2019. Unified Collaborative Filtering over Graph Embeddings. In *SIGIR*. 155–164.

[46] Pengfei Wang, Jiafeng Guo, Yanyan Lan, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2015. Learning Hierarchical Representation Model for NextBasket Recommendation. In *SIGIR*. 403–412.

[47] Zhe Xie, Chengxuan Liu, Yichi Zhang, Hongtao Lu, Dong Wang, and Yue Ding. 2021. Adversarial and Contrastive Variational Autoencoder for Sequential Recommendation. In *WWW*. 449–459.

[48] Haimo Zhang and Shengdong Zhao. 2011. Measuring Web Page Revisitation in Tabbed Browsing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. ACM, New York, NY, USA, 1831–1834. https://doi.org/10.1145/1978942.1979207

[49] Sai Qian Zhang, Qi Zhang, and Jieyu Lin. 2019. Efficient Communication in Multi-Agent Reinforcement Learning via Variance Based Control. In *NeurIPS 2019*. 3230–3239.

[50] Sai Qian Zhang, Qi Zhang, and Jieyu Lin. 2020. Succinct and Robust Multi-Agent Communication With Temporal Message Control. In *NeurIPS 2020*.

[51] Wayne Xin Zhao, Shanlei Mu, Yupeng Hou, Zihan Lin, Yushuo Chen, Xingyu Pan, Kaiyuan Li, Yujie Lu, Hui Wang, Changxin Tian, Yingqian Min, Zhichao Feng, Xinyan Fan, Xu Chen, Pengfei Wang, Wendi Ji, Yaliang Li, Xiaoling Wang, and Ji-Rong Wen. 2021. RecBole: Towards a Unified, Comprehensive and Efficient Framework for Recommendation Algorithms. In *CIKM*. 4653–4664.

[52] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. 2019. Deep Interest Evolution Network for Click-Through Rate Prediction. In *AAAI 2019*. 5941–5948.

[53] Guorui Zhou, Xiaoqiang Zhu, Chengru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep Interest Network for Click-Through Rate Prediction. In *KDD 2018*. 1059–1068.

[54] Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. 2020. S3-Rec: Self-Supervised Learning for Sequential Recommendation with Mutual Information Maximization. In *CIKM*. 1893–1902.

[55] Simiao Zuo, Haoming Jiang, Zichong Li, Tuo Zhao, and Hongyuan Zha. 2020. Transformer Hawkes Process. In *ICML 2020*, Vol. 119. 11692–11702.