Multi-Agent RL-based Information Selection Framework for Sequential Recommendation

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

Pengfei Wang*
Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia, Beijing University of Posts and Telecommunications
Beijing, China
wangpengfei@bupt.edu.cn

Chenliang Li*
School of Cyber Science and Engineering, Wuhan University
Wuhan, China
cllee@whu.edu.cn

ABSTRACT

For sequential recommender, the coarse-grained yet sparse sequential signals mined from massive user-item interactions have become the bottleneck to further improve the recommendation performance. To alleviate the spareness problem, exploiting auxiliary semantic features (*e.g.,* textual descriptions, visual images and knowledge graph) to enrich contextual information then turns into a mainstream methodology. Though effective, we argue that these different heterogeneous features certainly include much noise which may overwhelm the valuable sequential signals, and therefore easily reach the phenomenon of negative collaboration (*i.e.,* $1 + 1 < 2$). How to design a flexible strategy to select proper auxiliary information and alleviate the negative collaboration towards a better recommendation is still an interesting and open question. Unfortunately, few works have addressed this challenge in sequential recommendation.

In this paper, we introduce a **M**ulti-**A**gent **RL**-based **I**nformation **S**election Model (named MARIS) to explore an effective collaboration between different kinds of auxiliary information and sequential signals in an automatic way. Specifically, MARIS formalizes the auxiliary feature selection as a cooperative Multi-agent Markov Decision Process. For each auxiliary feature type, MARIS resorts to using an agent to determine whether a specific kind of auxiliary feature should be imported to achieve a positive collaboration. In between, a QMIX network is utilized to cooperate their joint selection actions and produce an episode corresponding an effective combination of different auxiliary features for the whole historical sequence. Considering the lack of supervised selection signals, we further devise a novel reward-guided sampling strategy to leverage exploitation and exploration scheme for episode sampling. By preserving them in a replay buffer, MARIS learns the action-value function and the reward alternatively for optimization. Extensive experiments on four real-world datasets demonstrate that our model obtains significant performance improvement over up-to-date state-of-the-art recommendation models.

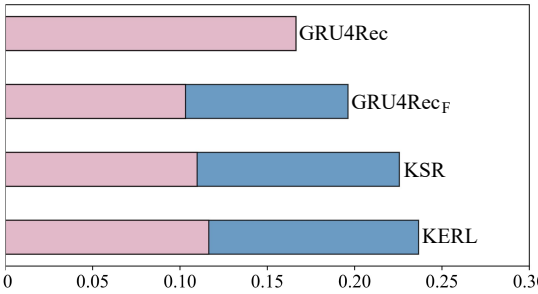

Figure 1: Performance comparison in terms of Recall@10 between GRU4Rec and its context-enhanced models (e.g., GRU4Rec$_F$, KSR, and KERL) on Beauty dataset. For each context-enhanced model, the pink bar represents the proportion that both GRU4Rec and its enhanced models recommended correctly, while the blue bar represents the enhanced models recommended correctly but GRU4Rec failed.

CCS CONCEPTS

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

KEYWORDS

sequential recommendation; multi-agent reinforcement learning; context-aware recommendation

ACM Reference Format:
Kaiyuan Li, Pengfei Wang, and Chenliang Li. 2022. Multi-Agent RL-based Information Selection Framework for Sequential Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22), July 11–15, 2022, Madrid, Spain.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3477495.3532022

1 INTRODUCTION

Recent years have witnessed the flourishing development of sequential recommendation [18, 42, 45]. In this task, sequential dependencies are critical signals that are proven to be useful [5, 8]. Following this line, many models are designed to extract sequential signals to infer users' purchase intentions for effective recommendation. However, as the collective behaviors of users are likely to be limited and fragmentary, these models usually suffer from data sparsity problem. These coarse-grained sequential patterns then become the

*Corresponding author

SIGIR '22, July 11–15, 2022, Madrid, Spain.

ACM ISBN 978-1-4503-8732-3/22/07…$15.00
https://doi.org/10.1145/3477495.3532022

bottleneck of further pushing the frontier of the recommendation performance.

To alleviate the sparsity problem, many research efforts have been devoted to exploit auxiliary features like textual descriptions[23], visual images[14] and knowledge graph [19]. A common paradigm of these solutions is to transform these heterogeneous information into generic feature vectors, which are fed into a supervised learning model [17] together with sequential representations to predict the user's next choice. Although these methods have provided promising performance, a critical deficiency is that their integration strategy may weaken the benefits of sequential signals. The noisy nature of these heterogeneous features and their adverse interference may finally result in a negative collaboration. To verify our assumption, we select one sequential recommender (e.g., GRU4Rec [16]), its several context-enhanced models (e.g., GRU4Rec$_F$ [17], KSR [19], and KERL [43]), and compare their performance on a real-world dataset (Beauty dataset from Amazon). The results are analyzed in Fig. 1.

We can see that these context-enhanced recommenders (*e.g.,* GRU4Rec$_F$ [17], KSR [19], and KERL [43]) perform better than GRU4Rec. However, more than 30% items that can be recommended successfully by GRU4Rec now failed in context-enhanced models. This obvious divergence between two types of recommenders demonstrates that the ineffective integration strategy inevitably overwhelms the benefits of sequential signals. Fig. 2 illustrates an example for this scenario. Here, we choose to exploit auxiliary information from three different sources. It is obvious to see that some auxiliary features are irrelevant to enhance the user's intent understanding, which could introduce significant semantic corruption. Hence, it is necessary to design a flexible selection strategy to select proper auxiliary information, and further guarantee their effective collaboration over sequential signals and different kinds of auxiliary features for better recommendation.

Although it is appealing in theory, it is non-trivial to realize an effective collaboration strategy without any supervision signal. Recently, some attention-based approaches are proposed [34, 48] to automatically assign different weights on each kind of auxiliary features in terms of their relevance estimation. However, the accumulation of potential noise may still significantly complicate the sequence learning process, as demonstrated in Fig. 1. How to automatically select different auxiliary information and further integrate them for better recommendation is an interesting and challenging problem.

In this work, we re-investigate the utility of different kinds of auxiliary information. Specifically, we propose a **M**ulti-**A**gent **RL**-based **I**nformation **S**election Model (named MARIS) to automatically select proper auxiliary feature combination that delivers an effective collaboration. In detail, we formalize the auxiliary feature selection as a cooperative Multi-agent Markov Decision Process. For each auxiliary information type, MARIS designs an agent to determine whether the information should be kept to enrich the contextual information of the historical sequence. Here, a QMIX Network is utilized to cooperate with their joint selection actions [32]. Considering the lack of supervised selection signals, we further devise a novel reward-guided sample strategy for optimization under an exploitation and exploration scheme: 1) MARIS first samples a series of potential high-quality episodes according to the QMIX network

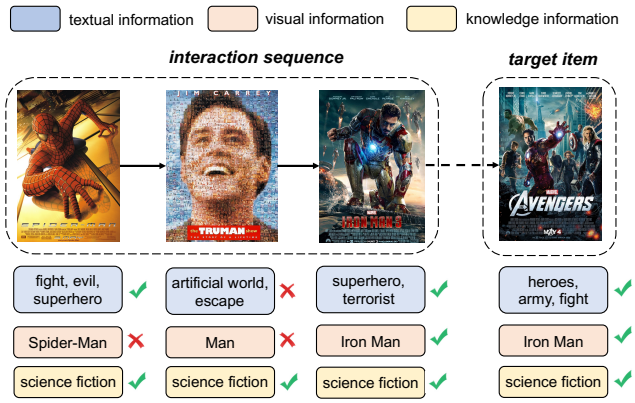

Figure 2: An example to illustrate the contextual information selection for better recommendation.

by a ranking function guided by the reward; 2) By keeping those high-quality episodes in a replay buffer, MARIS utilizes a Double Q-learning for efficient model optimization.

We construct extensive experiments on four datasets with a series of competitive baselines. Experimental results show that our proposed MARIS can significantly outperform all the baselines in multiple metrics. To summarize, the contributions of this paper are listed as follows:

- We formalize the auxiliary feature selection into a Multi-agent Markov Decision Process, and further utilize a QMIX network to cooperate their joint selection actions. To the best of our knowledge, this is the first work of considering the collaboration between different kinds of auxiliary information under a unified model.
- We propose a novel reward-guided sampling strategy to exploit and explore high-quality episodes. By maintaining episodes in a replay buffer, a Double Q-learning is utilized to guarantee the effectiveness and efficiency of the learning procedure.
- Extensive experiments on four real-world datasets demonstrate that our model obtains significant performance improvement over both the sequential and context-enhanced recommendation models.

2 RELATED WORK

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

Sequential Recommendation. Sequential recommendation strives to capture meaningful sequence patterns more efficiently. Early works mainly utilized Markov chain models [36, 44] to capture lower-order sequential dependencies. With the prosperity of deep models, Recurrent Neural Networks [16, 24, 27] and self-attention models [7, 20, 21, 38] have been adopted in several sequential modeling tasks to address the limitations in Markov models. For example, Hidasi et al.[16] applied Gated Recurrent Units (GRU) to model the whole session for a more accurate recommendation. Liu et al.

[27] proposed a short-term memory attention network by changing the recurrent encoder into an attention layer, which can further reduce the deviation in the time series. Recently, there are also some pre-training methods [28, 50] to derive the intrinsic data correlations for improving sequential recommendation. Though effective, these methods are usually challenged in handling limited user-item interactions.

Context-aware Recommendation. Context-aware Recommendation aims to leverage rich context information to improve the recommendation performance [40]. To fully exploit interactions among these features, previous works usually utilized matrix factorization technique [12, 14] for context modelling, such as libFM [35] and tensor factorization [6, 12]. Recently, deep learning techniques have also been utilized for context modelling. For example, Zhang et al. [47] integrated both structural, textual, and visual knowledge to jointly learn item representations. Wang et al. [41] modelled the evolution of different relations' effects with time, and incorporated such information into item embeddings. More recently, several studies also proposed to improve the sequential recommendation performance by integrating contextual information [25, 26, 34, 48]. Though effective, to the best of our knowledge, seldom works have considered the impact of sequential properties when introducing inappropriate contexts. A coarse fusion strategy may inevitably bring conflicts over sequential and contextual properties, and decrease the recommendation performance.

Multi-agent Reinforcement Learning. Multi-agent reinforcement learning (MARL) is a group of agents (or decision-makers) that interact with each other and their operating environment to achieve goals in a cooperative or competitive manner. MARL has made a breakthrough in recent years due to its ability to solve complex real-world problems, such as resource allocation in wireless networks, traffic signal control, flood monitoring, network routing, etc. These methods can largely be divided into policy-based and value-based methods [15]. Policy-based methods are promising for scaling to large action spaces, which try to maximize the future accumulated rewards by performing gradient ascent in policy space. For example, Lowe et al.[29] and Foerster et al.[11] typically use a centralised critic to estimate the gradient for a set of decentralised policies. Different from the policy-based methods, in value-based methods, value decomposition [4] is widely used. These methods learn individual Q-value functions for each agent, which are combined with a learnable mixing function to produce joint action values. For example, Sunehag et al.[39] utilized an arithmetic summation as the mixing function. Rashid et.al[32, 33] proposed a non-linear monotonic factorization structure. In our work, we aim to utilize MARL to leverage the collaboration between sequential and contextual properties.

3 PRELIMINARY

Notations. Let $\mathcal{U}$ denote a set of users and $\mathcal{I}$ denote a set of items, where $|\mathcal{U}|$ and $|\mathcal{I}|$ are the numbers of users or items. For each user $u \in \mathcal{U}$, we use $i^u_{1:t} = \{i^u_1, i^u_2, \cdots, i^u_t\}$ to represent the interaction sequence of items, where i^u_k represents the item that u has interacted with at k-th time step. In addition to users' interaction histories, we assume that there are totally m kinds of available auxiliary information, denoted as $C = \{c^1, c^2, \cdots, c^m\}$.

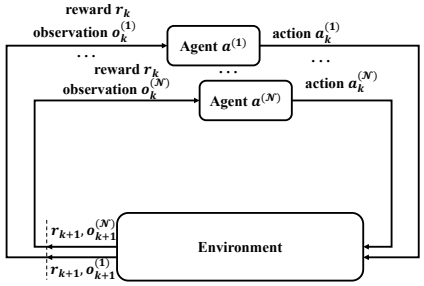

Figure 3: The overall architecture of MMDP.

Task Definition. Based on these notations, our task of *sequential recommendation* aims to predict the next item that the user u is likely to interact with at the $(t+1)$-th step given both the interaction sequence $i^u_{1:t}$ and the auxiliary information C.

Multi-agent Markov Decision Process. We first briefly introduce MMDP [3], and the framework of MMDP is shown in Fig. 3.

The MMDP can be described as a stochastic game G, represented as a tuple $G = \{\mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{T}, r, \gamma\}$, 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; $\mathcal{O} = \{O^{(1)}, O^{(2)}, \cdots, O^{(\mathcal{N})}\}$ is the set of observations; $\mathcal{A} = \{A^{(1)}, A^{(2)}, \cdots, A^{(\mathcal{N})}\}$ is the collection of action sets, with $a^{(j)}_k \in A^{(j)}$ being j-th agent's action at k-th time step; $\mathcal{T}$ is the state transition function: $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$; by their joint actions $a_k = \{a^1_k, a^2_k, \cdots, a^{\mathcal{N}}_k\}$ and discount factor γ, all agents share the same reward function $r(s_k, a_k)$.

4 METHODOLOGY

In this section, we introduce the proposed **M**ulti-**A**gent **R**L-based **I**nformation **S**election Framework (MARIS) in detail, and the overall architecture of MARIS is presented in Fig 4. In the following, we start with a Multi-Agent Markov Decision Process (MMDP) formulation for our task, then present our reward-guided sampling strategy for model optimization. For simplicity, we describe the approach for a single user, and we drop the superscript of u in the notations for ease of reading.

4.1 MMDP for Auxiliary Information Selection

We use MMDP to frame the auxiliary information selection process for sequential recommendation. In a MMDP, each agent is responsible for a specific auxiliary information type, aiming to interact with the environment at discrete time steps. We first adopt a standard Gated Recurrent Unit [16] to encode the previous item interaction sequence $i_{1:t}$, denoted as $\mathbf{h}^e_0$. Based on the initial $\mathbf{h}^e_0$, we aim to feed it to each agent to explore useful auxiliary information to enhance its semantics. Specifically, given agent j, the k-th step observation embedding can be written as:

$$\mathbf{o}^{(j)}_k = MLP(\mathbf{h}^e_{k-1} + \mathbf{c}^{(j)}_k) \tag{1}$$

where $\mathbf{c}^{(j)}_k$ is the embedding of auxiliary information $c^{(j)}_k$, $\mathbf{h}^e_{k-1}$ is the enhanced sequential representation at $(k-1)$-th step. Based

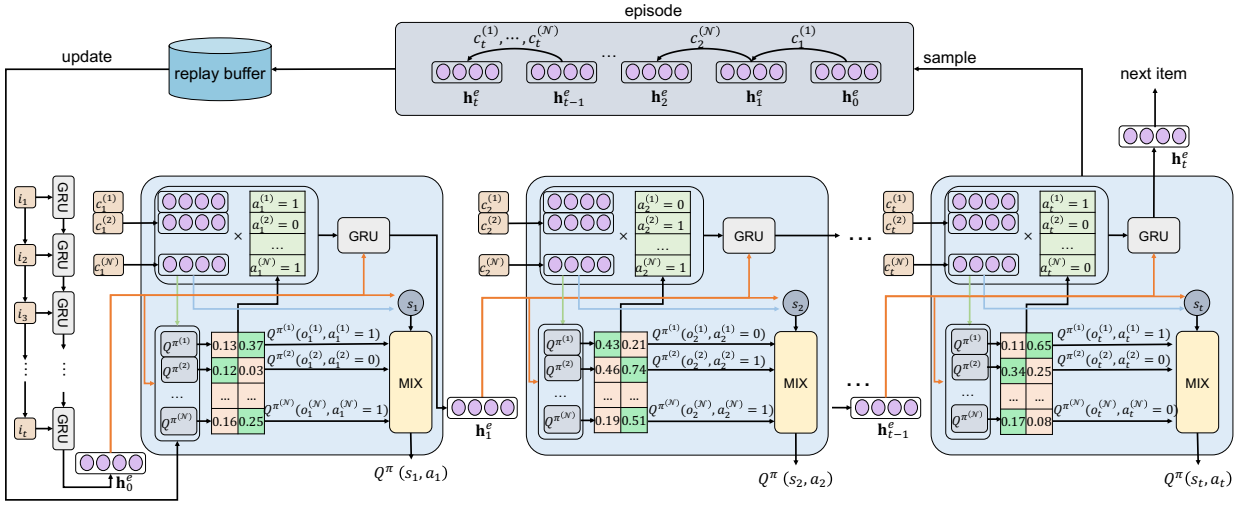

Figure 4: The overall architecture of Multi-Agent RL-based Information Selection Framework (MARIS for short). MARIS formalizes the auxiliary information selection process into a MMDP and utilizes a reward-guided sampling strategy to sample episodes for efficient optimization(Best viewed in color).

on the observation $\mathbf{o}_k^{(j)}$, we can calculate its action-value network $Q^{\pi^{(j)}}(o_k^{(j)}, a_k^{(j)})$ to make actions, where $\mathbf{a}_k^{(j)} \in \{\text{retain=1, neglect=0}\}$. In our work, the retain action indicates that the auxiliary information should be preserved, while a neglect action means that the auxiliary information should be discarded. According to such a design, our model can focus on the valuable auxiliary information, while avoiding the contamination of irrelevant auxiliary information.

After selecting proper auxiliary information at k-th step for each agent, we can use the following function to fuse proper auxiliary information into sequential representation to enhance its semantics:

$$\mathbf{h}_k^e = GRU\big(\mathbf{h}_{k-1}^e, \sum_j I(a_k^{(j)} = 1)\mathbf{c}_k^{(j)}\big) \tag{2}$$

where $I(\cdot)$ represents the identity function.

Up to now, we have designed decentralised policies for auxiliary information selection, which can operate effectively after giving a reward. However, such a design cannot explicitly represent interactions between the agents and may not converge, as each agent's learning is confounded by the learning and exploration of others. Inspired by [32], we further utilize QMIX to cooperate the actions centralised in an end-to-end fashion. The main advantage of QMIX is that it employs a network that estimates joint action-values as a complex non-linear combination over per-agent values. Specifically, we formulate the agent's state $\mathbf{s}_k$ as follows:

$$\mathbf{s}_k = MLP\left(\mathbf{h}_{k-1}^e + \sum_j \mathbf{c}_k^{(j)}\right) \tag{3}$$

Given the state s_k and the corresponding action-value of all agents, QMIX network utilizes the following function to obtain the joint action value $Q^{\pi}(s_k, a_k)$:

$$\begin{cases} \mathbf{z} = ELU\big(\mathbf{q} \cdot |MLP(\mathbf{s}_k)| + MLP(\mathbf{s}_k)\big) \\ Q^{\pi}(s_k, a_k) = \mathbf{z} \cdot |MLP(\mathbf{s}_k)| + MLP\Big(ReLU\big(MLP(\mathbf{s}_k)\big)\Big) \end{cases} \tag{4}$$

where $\mathbf{q} = [Q^{\pi^{(1)}}(o_k^{(1)}, a_k^{(1)}), \cdots, Q^{\pi^{(\mathcal{N})}}(o_k^{(\mathcal{N})}, a_k^{(\mathcal{N})})]$, $|\cdot|$ is an absolute operation to guarantee its weights non-negative, $ELU(\cdot)$ [9] and $ReLU(\cdot)$ [1] are activation functions.

According to Eq.4, QMIX can fully leverage the joint and individual action values, and then collaborate agents' actions for a better cooperation. Given the joint action-value, the loss function is written as:

$$\mathcal{L}(\Theta) = \sum_u \sum_{k=1}^{t} \Big[r(s_k, a_k) + \gamma \max_{a_{k+1}} Q^{\pi}(s_{k+1}, a_{k+1}) - Q^{\pi}(s_k, a_k); \Theta \Big]^2 \tag{5}$$

where Θ are all parameters in the learning space.

As in our task, we tend to give a high probability to a good enhanced sequential representation $\mathbf{h}_t^e$ that leads to the target item i_{t+1} together with its proper auxiliary information combination. To this end, we only consider to give a reward for the final state s_t. By enumerating all possible auxiliary information combinations of item i_{t+1}, we select the best matching with $\mathbf{h}_t^e$ as the terminal reward R_t via a softmax function:

$$R_t = \frac{e^{y_{t+1}}}{\sum_{i \in \mathcal{I}} e^{y_i}}; y_i = \max_{\mathcal{Y}_i^l \in \mathcal{Y}_i} \big(\mathbf{h}_t^e \cdot (\mathbf{v}_i + \sum_{c_i^{(j)} \in \mathcal{Y}_i^l} \mathbf{c}_i^{(j)})\big) \tag{6}$$

where $r(s_t, a_t)$=R_t, $\mathbf{v}_i$ denotes the embedding of the i-th item; $\mathcal{Y}_i$ denotes all the possible auxiliary information combination of the i-th item, $\mathcal{Y}_i^l \in \mathcal{Y}_i$. The total size of $\mathcal{Y}_i$ is $2^{\mathcal{N}}$.

4.2 Reward-guided Sampling Strategy

In the previous section we introduce MMDP to frame the auxiliary information selection process, and in the learning stage, we optimize our framework according to Eq. 5. However, in the learning stage, we found the performance of our model may not always be stable. The reason lies in the random episode sampling process. Due to the lack of supervised signals, the framework is difficult

to sample high-quality episodes. The low-quality episodes bring challenges for a better optimization. It is necessary to design an effective sampling strategy to assist its optimization. To this end, we aim to design a reward-guided sampling strategy of leveraging both exploration and exploitation for sampling episodes. By maintaining the high-quality episodes in a replay buffer, we learn our model efficiently.

Empower the sampling strategy to explore. We first encourage our model to explore more possibilities. In order to learn the action-value network effectively, it is necessary to explore episodes that are different from the previous sampled. Therefore, we utilize the following function to guarantee its exploration ability:

$$Rank_{explore}(\mathbf{h}_t^e) = 1 - \frac{(\mathbf{h}^* \cdot \mathbf{h}_t^e)}{|\mathbf{h}^*| \cdot |\mathbf{h}_t^e|} \tag{7}$$

where $\mathbf{h}_t^e$ is embedding of the current sampled episode, $\mathbf{h}^*$ is the representation of the episode stored in the replay buffer, which owns the highest ranking score. By this, we tend to give a high score to the sampled episode that is different with previous ones. According to such a design, we urge our model to explore more search space for effective optimization.

Empower the sampling strategy to exploit. Moreover, to gain the better performance of the recommendation, we hope that our model can well utilize previous high-quality episodes to make the learning process more efficiently, we then design an exploiting strategy written as follows:

$$Rank_{exploit}(\mathbf{h}_t^e) = R_t \times \frac{(\mathbf{h}^* \cdot \mathbf{h}_t^e)}{|\mathbf{h}^*| \cdot |\mathbf{h}_t^e|} \tag{8}$$

As we can see, such an exploiting strategy advocates our model to exploit episodes that are similar to the high-quality ones.

We can flexibly replace the cosine function with other forms of similarity measurements. By plugging Eq. 7 and Eq. 8, we compute the reward by considering episodes' both exploration and exploitation abilities. By plugging them together, we can derive the final sampling strategy:

$$Rank(\mathbf{h}_t^e) = \alpha \times Rank_{explore}(\mathbf{h}_t^e) + (1-\alpha) \times Rank_{exploit}(\mathbf{h}_t^e) \tag{9}$$

According to Eq. 9, we treat the reward R_t as the weakly supervised signal to direct the episode sampling process. By tuning the hyper-parameter α, we ensure a suitable trade-off between exploration and exploitation for an efficient episode sampling.

Algorithm 1 Learning algorithm for MARIS

Input: user-item interaction sequences, all auxiliary information for items, replay buffer , all parameters in the learning space Θ

```
Initialize Θ ← random values;
Initialize the replay buffer by two types of episodes.(one episode contains no auxiliary information, and the other one preserves all auxiliary information);
repeat
  for u in 𝒰 do
    Obtain h_0^e by encoding interaction sequence i_{1:t}^u
    for Sample-steps do
      Sample episodes according to QMIX
      Obtain h_t^e according to Eq. 2
      Calculate Rank(h_t^e) according to Eq. 9
      Rank episodes by Rank(h_t^e) and preserve top-n episodes in replay buffer
    end for
    for Train-steps do
      Sample each episode from the replay buffer
      for k=1 to t do
        Obtain s_k according to Eq.3
        Obtain Q^π(s_k, a_k) according to Eq.4
      end for
      Calculate R_t according to Eq. 6
      According to Eq. 5, learn the action-value and the reward R_t alternatively for optimization
    end for
  end for
until converge
return all parameters in Θ
```

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

Dataset	Beauty	CellPhones	Clothing	Movies
Users	22,363	27,879	39,387	123,960
Items	12,101	10,429	23,033	50,052
Interactions	198,502	194,439	278,677	1,697,533
Entities	14,422	11,590	25,322	51,830
Images	12,009	10,202	22,879	49,654
Texts	12,094	10,416	23,032	49,304
a.v.l	8.88	6.97	7.08	13.69

4.3 Learning and Discussion

Based on the reward-guided sampling strategy, our learning procedure is as follows: (1) For each user-item interaction sequence, we prepare a replay buffer to maintain its high-quality episodes. The replay buffer was first initialized by feeding two specific episodes: one episode contains no auxiliary information, while the other one keeps all auxiliary information, and their ranking scores are obtained according to Eq. 6. (2) Based on the introduced two episodes, we aim to find a series of compromise combination approaches of them. Specifically, by treating them as lower-bounds, we repeat the sampling process according to Eq. 9 to explore and exploit high-quality episodes. By ranking these episodes according to Eq. 9, we preserve top-n episodes in the replay buffer. Based on the sampled high-quality episodes. Given the loss function Eq. 5, we learn the reward R_t and the joint action-value $Q^\pi(\cdot)$ alternatively for optimization. The overall algorithm is given in Alg. 1.

The major novelty of the MARIS model lies in that MARIS formalizes the auxiliary information selection process into a MMDP, and a QMIX network is further introduced to coordinate their collaboration, such a factor has been missing in previous sequential recommendation models [26, 34], which may be challenged by the conflicts over properties when injecting various information. In addition, MARIS designs a reward-guided sampling strategy to leverage both exploration and exploitation processes for high-quality episodes.

In the recommendation procedure, with the learned MARIS, given a user and his/her interaction sequence, for each agent, we

Table 2: Performance comparison between baselines and MARIS (all values in the table are percentage numbers with % omitted). 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.

Dataset	Metric	Sequential Models			Context-aware Sequential Models						▲%
		FPMC	GRU4Rec	SASRec	GRU4Rec$_F$	KERL$_F$	SASRec$_F$	MFGAN	NOVA-BERT	MARIS	
Beauty	Recall@5	9.25	10.87	15.27	13.41	17.82	17.55	18.38	18.97*	**21.17**	11.60
	Recall@10	15.38	16.64	20.98	19.62	23.66	24.14	25.22	25.73*	**28.73**	11.65
	NDCG@5	6.25	7.05	10.96	9.19	11.59	11.98	12.26	13.66*	**14.93**	9.30
	NDCG@10	7.81	8.91	12.80	11.19	13.79	14.11	14.47	15.85*	**17.37**	9.59
CellPhones	Recall@5	13.70	14.77	17.69	15.59	18.40	18.55	19.36	20.82*	**24.30**	16.71
	Recall@10	20.02	20.74	24.16	23.12	26.02	26.13	26.73	29.28*	**32.92**	12.43
	NDCG@5	9.13	10.20	11.71	10.46	12.64	13.23	13.56	14.36*	**17.24**	20.06
	NDCG@10	11.16	12.14	14.36	12.88	15.60	15.40	15.93	17.09*	**20.02**	17.14
Clothing	Recall@5	5.58	6.10	7.62	8.24	11.14	12.74	13.21	15.15*	**16.69**	5.41
	Recall@10	8.47	9.42	10.73	12.66	16.86	18.62	20.10	21.06*	**23.91**	13.53
	NDCG@5	3.74	4.26	5.34	5.45	7.63	8.51	9.06	10.51*	**11.53**	9.51
	NDCG@10	5.12	4.89	6.34	6.87	10.38	11.34	11.88	12.62*	**14.15**	12.12
Movies	Recall@5	25.19	26.89	28.56	28.77	30.77	30.15	31.37	33.15*	**37.08**	11.86
	Recall@10	34.44	35.37	37.22	38.53	40.53	39.11	41.10	42.61*	**46.68**	9.55
	NDCG@5	17.48	19.57	20.79	20.23	21.94	22.21	23.17	24.25*	**27.83**	14.76
	NDCG@10	20.47	22.31	23.58	23.39	25.09	25.60	26.32	27.31*	**30.94**	13.29

scan each item and select the corresponding action according to the following function:

$$a_k^{(j)} = \arg\max_a Q^{\pi^{(j)}}(o_k^{(j)}, a) \tag{10}$$

After this, we aggregate the enhanced sequential representation $\mathbf{h}_t^e$ according to Eq. 2. Based on the learned $\mathbf{h}_t^e$, we rank the items according to Eq. 6, and select the top-N results as the final recommendations.

5 EXPERIMENT

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

5.1 Experimental Setup

Dataset. We conduct our experiments on the commonly-used Amazon dataset concerning its rich auxiliary information. To analyze our model's capability, we select four different categories, including Beauty, Clothing, Cell Phones and Movies. For these categories, we remove users and items with fewer than 5 related actions. The statistics of four datasets are shown in Table 1. We follow [47] and consider three different types of auxiliary information, which are visual, textual, and knowledge information of items. For the textual information, we consider both the titles and descriptions of items.

Baselines.To evaluate the effectiveness of our approach, We compare MARIS against two types of baselines, including three sequential-based models and five context-enhanced sequential models. The sequential-based models include:

(1) FPMC [36]: FPMC is a shallow model that combines matrix factorization and factorized first-order Markov chains for sequential recommendation.
(2) GRU4Rec [16]: GRU4Rec is a session-based recommendation, which utilizes GRU unit to capture users' long sequential behaviors for recommendation.
(3) SASRec [21]: SASRec is a self-attention based sequential recommendation model, which uses the multi-head attention mechanism to recommend the next item.

For context-aware sequential models, we consider the following five baselines:

(1) GRU4Rec$_F$ [17]: proposes to incorporate auxiliary information into GRU networks for improving the sequential recommendation. We concatenate the pre-trained auxiliary vectors and item embeddings as the input of GRU.
(2) SASRec$_F$: Similar to GRU4Rec$_F$, we extend SASRec with the concatenation of item embeddings and the pre-trained auxiliary information representations.
(3) KERL$_F$: KERL[43] is a knowledge-enhanced model for sequential recommendation, and we extend it by replacing the kg information with the pre-trained auxiliary information.
(4) MFGAN [34]: MFGAN designs a multi-discriminator structure that can decouple different auxiliary information to improve the recommendation performance.
(5) NOVA-BERT [26]: NOVA-BERT uses a non-invasive self-attention mechanism to make use of side information for a better recommendation.

Evaluation Metric. In order to present a comprehensive evaluation, for each user, we sort his records according to the timestamp to form the interaction sequence. Based on the sorted sequences, we hold out the last item of each sequence as the test data and the penultimate item of each sequence as the validation data. The rest data is treated as the training data. We set 1,000 negative items for each ground-truth item considering both the computation efficiency [43] and the estimation quality [22].

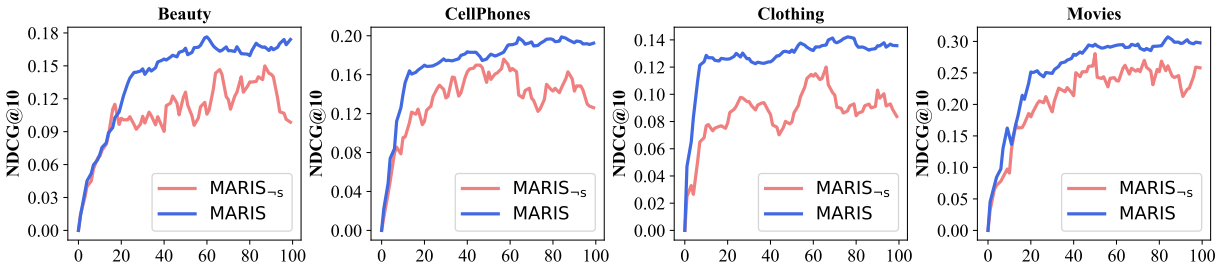

Figure 5: Performance curves of MARIS and its variant MARIS$_{\neg s}$ with the varying iterations on four datasets.

We employ the commonly used Recall@N, NDCG@N as our evaluation metrics(N=5/10). Recall@N measures the percentage of target items appearing in the top-N results, and NDCG@N takes the ranking position in the top-N list into account. 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 fair comparison, we adopt the following settings for all methods: the batch size is set to 256; all embedding parameters are randomly initialized in the range of (0, 1); the model dimension is tuned in the range of[32, 64, 96, 128, 256]. For KERL[1] and MFGAN [2], we use the source code provided by their authors. For other methods, we implement them by RecBole [49]. We optimize them according to the validation sets.

For our model, we implement it based on PyMARL[37]. The discount factor γ is set to 0.99, α is set to 0.4. In the sampling stage, we preserve Top-10 episodes in our replay buffer for each sequence. For textual information, we train the words according word2vec [31], and average them as the textual representation; for knowledge information, we use transE [2] to obtain their embeddings; for visual information, we apply PCA to reduce the initial embedding provided by amazon[13, 30]. The embedding size of all auxiliary information representations is set to 128.

5.2 Performance Comparison

In this section, we compare the performance of our model with the baselines. The overall performance of our proposed MARIS and the baselines are reported in Table 2. We have the following observations:

For sequential recommendations, FPMC obtains the worst performance. This is easy to understand as compared with other models, the shallow model FPMC neither fully utilizes sequential dependencies, nor injects extra auxiliary information to alleviate the sparse problem. Comparing with FPMC, we found that both considering sequential patterns (GRU4Rec) and utilizing the attention mechanism (SASRec) can improve the recommendation performance, Similar results have also been shown in previous works [10, 19].

After introducing the auxiliary information, both GRU4Rec$_F$ and SASRec$_F$ perform better than their initial models GRU4Rec and SASRec. It demonstrates the effectiveness of fusing side information to improve the recommendation performance. We find that there is no consistent dominant between KERL$_F$ and SASRec$_F$. This observation also reveals that exploring more auxiliary information and exploiting the enhanced sequential dependencies can both bring benefits in their own way. Comparing with the context-enhanced models GRU4Rec$_F$, SASRec$_F$, and KERL$_F$, MFGAN and NOVA-BERT take advantage of more complex strategies of fusing heterogeneous information, and achieve the better performance.

Finally, our proposed approach MARIS achieves the best performance among all the methods on four datasets. The major contribution of MARIS is that it considers a novel auxiliary information selection task. By formalizing the proposed task into a MMDP and further utilizing a QMIX network to coordinate collaboration among agents, MARIS designs a novel reward-guided sampling strategy for an efficient and effective optimization. Comparing with NOVA-BERT which utilizes a transformer to introduce proper auxiliary information for information aggregation, our MARIS utilizes a hard-attention strategy for auxiliary information selection. The result demonstrates that MARIS is more able to filter the irrelevant information, and cooperates this information for a better performance. Take the Cell Phones dataset as an example, when comparing with the best baseline (i.e., NOVA-BERT), the performance improvement of MARIS in terms of relative value is around 12.43% and 17.14% on Recall@10 and NDCG@10.

5.3 Ablation Study

In this section, we conduct experiments to analyze variants of MARIS via ablation study.

5.3.1 Analysis on QMIX network. Recall MARIS utilizes a QMIX network to coordinate the collaboration of different agents, in this section we aim to analyze whether such a design can bring benefits. Specifically, we directly remove the QMIX and sum the Q-value of each agent as the final joint action-value. According to such a design, our MARIS degrades to VDN [39], and we denote the new variant as MARIS$_{sum}$. The results of MARIS and MARIS$_{sum}$ on four datasets are shown in Table 3.

We can see that MARIS performs obviously better than MARIS$_{sum}$. It reveals that using a simple approach to coordinate the collaboration of multi agents is not an ideal choice. While owning to the QMIX network to coordinate agents' actions, our MARIS can directly leverage the joint and individual action-values to coordinate their collaboration for a better performance.

5.3.2 Analysis on Reward-guided sampling strategy. In MARIS, we design a reward-guided sampling strategy to obtain high-quality

[1]https://github.com/fanyubupt/KERL
[2]https://github.com/ReyonRen/MFGAN

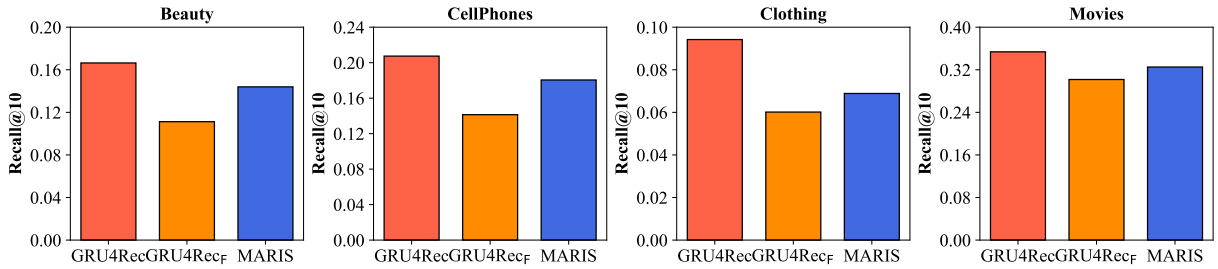

Figure 6: Comparisons between GRU4Rec and its two enhanced models. The red bar represents the performance of GRU4Rec. The purple bar represents the overlap between GRU4Rec and GRU4Rec$_F$, while the blue grid represents the overlap between GRU4Rec and MARIS.

Table 3: Performance comparison of MARIS and MARIS$_{sum}$ over four datasets. All numbers in the table are percent numbers with % omitted. The Best performance is in bold font.

Dataset	Metrics	MARIS$_{\text{sum}}$	MARIS
Beauty	Recall@5	18.51	**21.17**
	Recall@10	25.34	**28.73**
	NDCG@5	12.40	**14.93**
	NDCG@10	14.68	**17.37**
CellPhones	Recall@5	21.84	**24.30**
	Recall@10	29.27	**32.92**
	NDCG@5	14.87	**17.24**
	NDCG@10	17.49	**20.02**
Clothing	Recall@5	15.19	**16.69**
	Recall@10	21.84	**23.91**
	NDCG@5	10.12	**11.53**
	NDCG@10	12.40	**14.15**
Movies	Recall@5	35.07	**37.08**
	Recall@10	44.34	**46.68**
	NDCG@5	26.07	**27.83**
	NDCG@10	28.98	**30.94**

episodes for optimization. To verify the effectiveness of such a sampling strategy, we also make some degradation of MARIS. Specifically, for each interaction sequence, without using Eq. 9, we sample episodes randomly to fill the replay buffer. We then use these episodes to learn our model. The new variant of MARIS is named MARIS$_{\neg s}$. Fig. 5 shows the performance curve of MARIS and MARIS$_{\neg s}$ on four datasets.

We can see that comparing with MARIS, MARIS$_{\neg s}$ is difficult to converge on all four datasets. It is easy to understand that a random sampling strategy is difficult to obtain valuable episodes due to the huge search space, the low-quality episodes then bring difficulties for a robust optimization. While comparing with MARIS$_{\neg s}$, MARIS performs better and converges faster. It demonstrates the correctness and necessity of the reward-guided sampling strategy in MARIS. By leveraging the exploration and exploitation scheme to sample episodes, MARIS maintains a series of well-ranked episodes in the replay buffer, forcing the model to optimize efficiently.

5.4 Effect of the Hyper-parameter α

In MARIS we adopt a reward-guided sampling strategy for efficient learning of the proposed MARIS. One important parameter in this procedure is the hyper-parameter α that leverages the exploration and exploitation scheme. In this experiment, we study the impact of the α on the final performance. Specifically, We vary the value of α from 0 to 1 on the Movies dataset, and Fig. 7 shows the testing performance of MARIS in terms of NDCG@10 against α on Movies dataset.

We can see that when α=1, MARIS achieves the worst performance. It demonstrates that a pure exploration strategy does not fit MARIS, exploring too many low-quality episodes cannot drive MARIS to obtain a better performance. While α decreases, the testing performance of MARIS in terms of NDCG@10 increases. However, if we further decrease α, the overall performance decreases. When α=0, MARIS emphasizes on exploiting the episodes that are similar to the previous ones. Exploring a small search space also fails to learn MARIS well. These observations verify the underlying intuition of our model design, where we need an appropriate weighting value to balance the exploration and exploitation scheme. Therefore, to trade-off between these two factors, we set α=0.4 for the best performance in the experiments.

5.5 Feeding Other Sequential Recommendation Models to MARIS

In MARIS, the initial sequential representation $\mathbf{h}_0^e$ can be obtained by other recommendation models. In this section, we conduct experiments to check whether MARIS can obtain further improvements when merging other models. Specifically, we select four widely used sequential recommenders including DREAM[46], NARM[24], STAMP[27], and SASRec[21]. For each model, we obtain its sequential representation, and further feed it to our MARIS to analyze the performance of MARIS.

The results of NDCG@10 on Movies dataset are illustrated in Figure 8. We can see that by replacing the initial representation

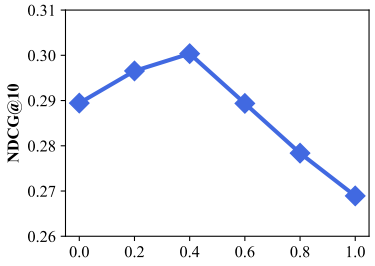

Figure 7: Performance variation in terms of NDCG@10 against α on the Movies dataset, where α varies from 0 to 1.

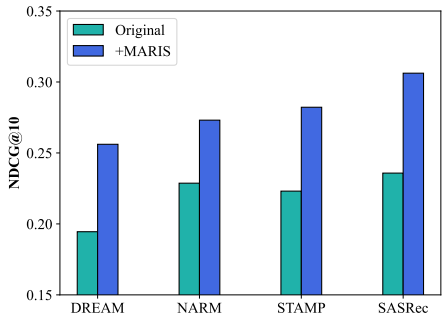

Figure 8: Performance of MARIS on Movies dataset when considering different sequential models. The green grids represent the performance of original sequential models, while the blue grids indicate the performance of MARIS after injecting the corresponding sequential models.

$\mathbf{h}_0^e$ to other sequential models, the performance of MARIS are still improved. It demonstrates the effectiveness and flexibility of our model, which can obtain benefits when combining with various of sequential recommendations.

5.6 Further Analysis on MARIS

In MARIS we formalize the auxiliary information selection task into a MMDP, and utilize a QMIX network to coordinate their actions. In this section, we further analyze the negative effects that the injected auxiliary information brings to sequential dependencies in MARIS. Considering MARIS uses a GRU Unit to aggregate its sequential representation, we then select GRU4Rec$_F$ for a fair comparison. According to compare the overlaps between GRU4Rec and its two context-enhanced models (e.g. GRU4FRec$_F$ and MARIS), we want to check whether MARIS would alleviate the negative collaboration. The result is shown in Fig. 6.

We see that the overlap between GRU4Rec and MARIS is obviously larger than the overlap between GRU4Rec and GRU4Rec$_F$. Take the Beauty dataset as an example, we can see that MARIS captures 87.3% of instances that GRU4Rec can recommend correctly, while for GRU4Rec$_F$, the proportion decreases to 64%. The huge gap between MARIS and GRU4Rec demonstrates the effectiveness of our MARIS, which shows a good generalization ability in alleviating the negative collaboration over different information.

5.7 Visualization Analysis

In this section, we analyze the significance of different combination types for a correct recommendation over each dataset, to understand how MARIS conducts automatic auxiliary information selection by coordinating the actions of agents. Specifically, for each interaction sequence that MARIS recommends correctly in the testing set, we statistic the auxiliary information combination type of each item in the interaction sequence. In our work we consider the selection over 3 different auxiliary information types, thus we have 8 combination types. Based on these types, we calculate their percentages. The percentage distributions on four datasets are shown in Fig. 9.

As we can see, nearly 40% of items select all three auxiliary types, and the result is quite consistent on all four datasets. It demonstrates

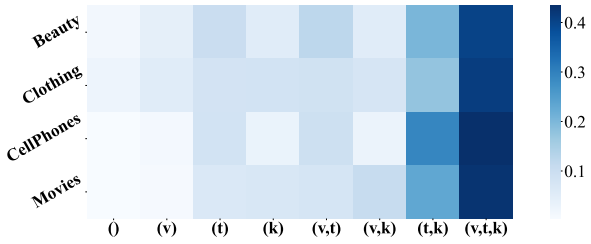

Figure 9: Distribution of different auxiliary information combination types on four datasets. The x-axis denotes the different context combination types, t stands for textual information, v stands for visual information, and k represents the knowledge. Each cell indicates the frequency of its corresponding combination type.

the importance of auxiliary information, which also coincides with the previous findings: introducing auxiliary information is a benefit for improving the recommendation performance. However, we find more than 25% of items choose visual and knowledge information as to their best matching combinations. For the rest 35% items, they distribute evenly on the other rest six combination types. The diverse distribution on different selection types demonstrates not all auxiliary information are needed for recommendation. It is necessary to make a precise selection over different auxiliary information. Overall, experimental results imply that MARIS is able to coordinate the collaboration over agents so as to conduct automatic model selection for sequential recommendation scenarios.

6 CONCLUSION

In this paper, we address an auxiliary information selection task in sequential recommendation scenario. We formalize this task into a MMDP, and propose a **M**ulti-**A**gent **RL**-based **I**nformation **S**election Model (MARIS for short) to explore an effective collaboration between different kinds of auxiliary information and sequential signals in an automatic way. MARIS utilizes a QMIX network to model the complex collaboration among various properties. After this, a reward-guided sampling strategy is further designed to leverage both exploration and exploitation scheme of sampling high-quality episodes for optimization. With the ranked episodes maintained in the replay buffer, MARIS optimizes the model effectively. Experiments on four various datasets verify the effectiveness of our proposed model.

To our knowledge, it is the first time of considering auxiliary information selection in sequential recommendation task. Currently, our focus only considers the collaboration among three different auxiliary information, we then coordinate their actions according to a simple QMIX network, and there is much work to be done. In the future, we aim to inject more auxiliary information, and further utilize weakly supervised signals to better understand interactions among these heterogeneous information for further improvement.

ACKNOWLEDGMENTS

This research work was supported by fundamental Research for the National Natural Science Foundation of China (No.61802029, 61872278). We would like to thank the anonymous reviewers for their valuable comments.

REFERENCES

[1] Abien Fred Agarap. 2018. Deep Learning using Rectified Linear Units (ReLU). *CoRR* abs/1803.08375 (2018).

[2] Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-relational Data. In *NIPS*. 2787–2795.

[3] Craig Boutilier. 1996. Planning, Learning and Coordination in Multiagent Decision Processes. In *Proceedings of the Sixth Conference on Theoretical Aspects of Rationality and Knowledge, De Zeeuwse Stromen, The Netherlands, March 17-20 1996*. 195–210.

[4] Jacopo Castellini, Frans A. Oliehoek, Rahul Savani, and Shimon Whiteson. 2019. The Representational Capacity of Action-Value Networks for Multi-Agent Reinforcement Learning. In *AAMAS*. 1862–1864.

[5] Jianxin Chang, Chen Gao, Yu Zheng, Yiqun Hui, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. 2021. Sequential Recommendation with Graph Neural Networks. In *SIGIR*. 378–387.

[6] Huiyuan Chen and Jing Li. 2019. Adversarial tensor factorization for context-aware recommendation. In *RecSys*. 363–367.

[7] Junsu Cho, SeongKu Kang, Dongmin Hyun, and Hwanjo Yu. 2021. Unsupervised Proxy Selection for Session-based Recommender Systems. In *SIGIR*. 327–336.

[8] Minjin Choi, Jinhong Kim, Joonseok Lee, Hyunjung Shim, and Jongwuk Lee. [n.d.]. Session-aware Linear Item-Item Models for Session-based Recommendation. In *WWW, pages = 2186–2197, year = 2021,*.

[9] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. 2016. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). In *ICLR*, Yoshua Bengio and Yann LeCun (Eds.).

[10] Zeyu Cui, Yinjiang Cai, Shu Wu, Xibo Ma, and Liang Wang. 2021. Motif-aware Sequential Recommendation. In *SIGIR*. 1738–1742.

[11] Jakob N. Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. 2018. Counterfactual Multi-Agent Policy Gradients. In *AAAI*. 2974–2982.

[12] Hancheng Ge, James Caverlee, and Haokai Lu. 2016. TAPER: A Contextual Tensor-Based Approach for Personalized Expert Recommendation. In *RecSys*. 261–268.

[13] Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *WWW*. 507–517.

[14] Ruining He and Julian J. McAuley. 2016. VBPR: Visual Bayesian Personalized Ranking from Implicit Feedback. In *AAAI*. 144–150.

[15] Pablo Hernandez-Leal, Bilal Kartal, and Matthew E. Taylor. 2019. A survey and critique of multiagent deep reinforcement learning. *AAMAS* 33, 6 (2019), 750–797.

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

[17] Balázs Hidasi, Massimo Quadrana, Alexandros Karatzoglou, and Domonkos Tikk. 2016. Parallel Recurrent Neural Network Architectures for Feature-Rich Session-Based Recommendations. In *RecSys*. 241–248.

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

[20] Taegwan Kang, Hwanhee Lee, Byeongjin Choe, and Kyomin Jung. 2021. Entangled Bidirectional Encoder to Autoregressive Decoder for Sequential Recommendation. In *SIGIR*. 1657–1661.

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

[22] Walid Krichene and Steffen Rendle. 2020. On Sampled Metrics for Item Recommendation. In *KDD*. 1748–1757.

[23] Chenliang Li, Xichuan Niu, Xiangyang Luo, Zhenzhong Chen, and Cong Quan. 2019. A Review-Driven Neural Model for Sequential Recommendation. In *IJCAI*. International Joint Conferences on Artificial Intelligence Organization, 2866–2872.

[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] Yang Li, Tong Chen, Peng-Fei Zhang, and Hongzhi Yin. 2021. Lightweight Self-Attentive Sequential Recommendation. In *CIKM*. 967–977.

[26] Chang Liu, Xiaoguang Li, Guohao Cai, Zhenhua Dong, Hong Zhu, and Lifeng Shang. 2021. Non-invasive Self-attention for Side Information Fusion in Sequential Recommendation. In *AAAI*.

[27] 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.

[28] Zhiwei Liu, Ziwei Fan, Yu Wang, and Philip S. Yu. 2021. Augmenting Sequential Recommendation with Pseudo-Prior Items via Reversely Pre-training Transformer. In *SIGIR*. 1608–1612.

[29] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. 2017. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. In *NIPS*. 6379–6390.

[30] Julian J. McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. 2015. Image-Based Recommendations on Styles and Substitutes. In *SIGIR*. 43–52.

[31] Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *NIPS*. 3111–3119.

[32] Tabish Rashid, Mikayel Samvelyan, Christian Schröder de Witt, Gregory Farquhar, Jakob N. Foerster, and Shimon Whiteson. 2018. QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. In *ICML*. 4292–4301.

[33] Tabish Rashid, Mikayel Samvelyan, Christian Schröder de Witt, Gregory Farquhar, Jakob N. Foerster, and Shimon Whiteson. 2020. Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. *JMLR* 21 (2020), 178:1–178:51.

[34] Ruiyang Ren, Zhaoyang Liu, Yaliang Li, Wayne Xin Zhao, Hui Wang, Bolin Ding, and Ji-Rong Wen. 2020. Sequential Recommendation with Self-Attentive Multi-Adversarial Network. In *SIGIR*. 89–98.

[35] Steffen Rendle. 2012. Factorization Machines with libFM. *ACM Trans. Intell. Syst. Technol.* 3, 3, Article 57 (May 2012), 22 pages.

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

[37] Mikayel Samvelyan, Tabish Rashid, Christian Schroeder de Witt, Gregory Farquhar, Nantas Nardelli, Tim G. J. Rudner, Chia-Man Hung, Philiph H. S. Torr, Jakob Foerster, and Shimon Whiteson. 2019. The StarCraft Multi-Agent Challenge. *CoRR* (2019).

[38] 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.

[39] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinícius Flores Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z. Leibo, Karl Tuyls, and Thore Graepel. 2018. Value-Decomposition Networks For Cooperative Multi-Agent Learning Based On Team Reward. In *AAMAS*. 2085–2087.

[40] Norha M. Villegas, Cristian Sánchez, Javier Díaz-Cely, and Gabriel Tamura. 2018. Characterizing context-aware recommender systems: A systematic literature review. *Knowledge-Based Systems* 140 (2018), 173–200.

[41] Chenyang Wang, Min Zhang, Weizhi Ma, Yiqun Liu, and Shaoping Ma. 2020. Make It a Chorus: Knowledge- and Time-aware Item Modeling for Sequential Recommendation. In *SIGIR*. 109–118.

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

[43] Pengfei Wang, Yu Fan, Long Xia, Wayne Xin Zhao, ShaoZhang Niu, and Jimmy Huang. 2020. KERL: A Knowledge-Guided Reinforcement Learning Model for Sequential Recommendation. In *SIGIR*. 209–218.

[44] 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.

[45] 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.

[46] Feng Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. A Dynamic Recurrent Model for Next Basket Recommendation. In *SIGIR*. 729–732.

[47] Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. 2016. Collaborative Knowledge Base Embedding for Recommender Systems. In *KDD*. 353–362.

[48] Tingting Zhang, Pengpeng Zhao, Yanchi Liu, Victor S. Sheng, Jiajie Xu, Deqing Wang, Guanfeng Liu, and Xiaofang Zhou. 2019. Feature-level Deeper Self-Attention Network for Sequential Recommendation. In *IJCAI*. 4320–4326.

[49] 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.

[50] 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.