A DQN-BASED POLICY

For the policy used in the experiments, we use a basic DQN policy with a Gated Recurrent Unit (GRU)-based network to encode discrete state and approximate action-value function. We follow the DQN paradigm and approximate the optimal action-value function $Q(s,a)$, to maps a state-action pair to the expected discounted cumulative reward $\sum_t \gamma^t r_t$ under the optimal policy, using a deep neural network $\hat{Q}(s,a;\theta) \approx Q(s,a)$.

The state encoder with a GRU-based network involves the following layers: (i) Embedding Look-up layer: Look-up the embeddings of the historical interaction items $[i_1, i_2, \ldots, i_t]$ and corresponding feedbacks $[f_1, f_2, \ldots, f_t]$ in state s_t from an embedding layer, denoted as $[\boldsymbol{v}_{i_1}, \boldsymbol{v}_{i_t}, \ldots, \boldsymbol{v}_{i_t}]$ and $[\boldsymbol{v}_{f_1}, \boldsymbol{v}_{f_t}, \ldots, \boldsymbol{v}_{f_t}]$. (ii) Embedding Combination: Combine the embeddings of items and feedback by using element-wise multiplication, denoted as $[\boldsymbol{v}_{i_1} \circ \boldsymbol{v}_{f_1}, \boldsymbol{v}_{i_2} \circ \boldsymbol{v}_{f_2} \circ \cdots \circ \boldsymbol{v}_{i_t} \circ \boldsymbol{v}_{f_t}]$. (iii) A GRU Layer: Use a GRU layer to compute the embedding of state: $\boldsymbol{h}_t = \mathrm{GRU}(\boldsymbol{h}_{t-1}, \boldsymbol{v}_{i_t} \circ \boldsymbol{v}_{f_t}; \Theta^{\mathrm{GRU}})$, where $\mathrm{GRU}(\cdot)$ is the GRU unit [9] with activation *tanh*, and Θ^{GRU} denotes the parameters of this GRU layer. Then we use a feedforward layer: map the embedding of state $\boldsymbol{h}_t$ into a vector as Q-value $Q(s,a)$ containing the Q value for any action a. The *behavior network* including the above-mentioned four-layer network is used to estimate Q-value function $\hat{Q}(s_t, a; \theta)$, where θ represents parameters of the *behavior network*, including all parameters of the above four layers.

To stabilize the training process, DQN introduces a *behavior network* separate from the *target network*. The *target network* is structured in the same way and share the embedding layer with the *behavior network*. The *target network* estimates the target-Q value function $\hat{Q}'(s,a;\theta')$, with the parameters θ' fixed and periodically copied from the *behavior network*. The *target network* is used to calculate Q-values $Q'(s_{t+1}, *)$ for destination state s_{t+1} using a forward pass through the target network. The parameters θ of the *behavior network* are updated by minimizing the following differentiable loss functions with the *Adam* optimizer,

$$L(\theta) = \mathbb{E}_{(s_t, a_{t+1}, r_{t+1}, s_{t+1})}[(r_{t+1} + \gamma \max_{a'} Q'(s_{t+1}, a'; \theta') - Q(s_t, a_{t+1}; \theta))^2], \tag{14}$$

where θ and θ' include shared embeddings, private parameters of GRU layer and feedforward layer in the behavior network and the target network respectively. where the parameters θ' of the *target network* are not updated in each learning step, but replaced by θ after multiple learning steps. We implement this DQN with the library of Tensorflow by using a *Adam* optimization in mini-batches.

B HYPERPARAMETERS

Table 2. List of Hyperparameters and their values.

Hyperparameter	Definition	Value		
		Yahoo!R3	Coat	Synthetic
Memory Size	The number of transitions stored in the replay memory.	20000	6000	6000
Discount factor	Discount fator γ used in the DQN.	0.9	0.9	0.9
Learning rate	The learning rate used by *Adam* optimization.	1e-4	1e-4	1e-4
Lr decay frequency	The number of step with which learning rate plus 0.9.	20000	5000	10000
Epsilon	The minimal probability of recommending an item randomly when taking an action.	0.1	0.1	0.1
Epsilon decay frequency	The number of step with which the epsilon ϵ (initial value as 0.8) minus 0.1.	20000	10000	20000
Minibatch size	The number of training cases randomly selected from replay memory and being used to update the parameters of policy.	512	128	128
Targetnet replacement frequency	The number of step with which the target network is updated.	100	20	20
Embedding dim	The dimension of Embedding Look-up layer.	100	100	50
GRU state dim	The dimension of state $\boldsymbol{h}_t$ encoded by the GRU layer.	100	10	10