# Online DRQN-Based E-Commerce Recommendation

**Xiaoyi Cai**
**Faculty of Applied Mathematics and Computer Sciences, Belarusian State University, Minsk 220030, Belarus.**

***Abstract:*** E-commerce recommendation systems, which have the advantage of dynamic interaction over traditional recommendation algorithms, use reinforcement learning for virtual human-e-commerce interaction to model the interests of users. In this paper, we first embed users and items, and then pass them through an LSTM network. Unlike DRQNs, we do not require image recognition, so we do not use convolutional neural networks and instead only consider training these RL-based recommender systems via LSTM in a RecoGym environment [1]. In this paper, we use RecoGym to generate artificial data, rather than real data, in consideration of the requirement to protect user privacy and to be more easily compatible with a reinforcement learning environment. A combination of Long Short Term Memory (LSTM) and DQN is deployed. This compensates for the fact that DQNs handle longer sequences of actions or achieve "relaxed Markovian" learning across sequences. Similar modelling effects were demonstrated in [2][3][4], but this paper uses a combination of reinforcement learning algorithms and recurrent neural networks and achieves better error convergence rates and recommends items to users more efficiently and accurately.

***Keywords:*** Recommender System; DQN; LSTM; Online E-Commerce

## Introduction

With the growth of the Internet, mobile has advanced. This has led to an increasing demand for recommendation systems. Today, recommendation algorithms are used in movie recommendation, news recommendation, e-commerce product recommendation, advertising recommendation and many other fields.The goal of reinforcement learning recommendation is to train and predict the future behavior of a user based on the user's historical action sequences interacting with an agent.The mathematical model underlying reinforcement learning is the Markov decision process[5].

But the pure reinforcement learning model, suffers from the drawback of processing longer sequences, Markov chains (MCs), which assume that the next action depends only on the previous action (or previous ones). while the higher-order Markov will lead to spatial problems. Existing recommendation systems focus on short-term interest modeling, estimating the immediate response of users to recommendationsTherefore, in this paper, we combine DQN with LSTM to make the intelligence possess memory, solving limited memory for experience storage,which will help us to handle longer sequences,Capturing the long-term interest of users.

## Methodology

## Deep Q-learning

DQN is a learning control strategy for intelligences that interact with an unknown environment. These environments can be formalized as Markov Decision Processes (MDPs).

Generally, every MDP are described by a tuple $\langle S, A, P, R, \gamma \rangle$ and the application on e-commerce recommendation systems can be defined as:

**State space, S:** State is about a record of user behaviour generated offline with recogym. set{u,commodities,timestampe}.

**Action space, A:** Recommend items recommended by the system at the time of action,{commodities}.

**Reward, R:** The reward is the user's feedback (click or not) on the recommended item,The agent is immediately rewarded based on the user's feedback r(s,a).

**Transition probability, P:** is the probability of a state change from st to st+1. This conditional probability is assumed to satisfy the Markov property. Simply put, we can predict the future based on the present based on this property.

**Discount factor, γ**:This value is a portrayal of the relative importance of future rewards compared to immediate rewards. If γ = 0, the agent pays attention only to immediate rewards and, conversely, if γ = 1, the algorithm considers all future rewards for taking the current action.

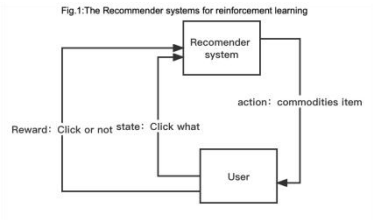



Fig.1:It is Shows the process of interaction between the recommender system (RL intelligence) and the user and the flow of RL.Our goal is to maximise the Q(s, a) value when the user interacts with the recommender system to produce a Q(s, a) value, thus achieving a higher CTR.

It is can be defined as[3]:

$$Q(s,a) = Q(s,a) + \alpha[r + \gamma max_{\alpha} Q_(s_i', a') - Q(s,a)] \quad (1)$$

Updating the parameters of the network to minimise the differentiable loss function can be defined as[3]:

$$L(s,a|\theta_i) = (r + \gamma max_{a'} Q(s', a'|\theta_i) - Q(s,a|\theta_i))^2 \quad (2)$$

$$\theta_{i+1} = \theta_i + \alpha \nabla_{\theta} L(\theta_i) \quad (3)$$

# POMDP

In CosRec: 2D convolutional neural networks for sequential

Recommendations paper [2], a non-strict Markov chain that is shown to optimise recommenders. As an example, in a strict Markov chain, the current action is only correlated with the previous action, i.e. inside the recommender system, the user clicks on the camera, the memory card, the lens, so that a strict correlation can recommend the tripod. But if we relax the conditions, say we click on camera, memory card, but then click on bike, and then click on lens, the system will still recommend the tripod item to us. As can be seen, there are many partial Markovian examples of user behaviour in the e-commerce domain, where the likelihood of the next product varies considerably given the different patterns of time intervals between past user events. So in this paper, we add a recursive deep Q network to the DQN, which better titrates the actual Q values of the observed sequences.

# Experiment

**Dataset:**The data is generated manually, unlike real user data, and is more privacy-protective.

| Model | DQN | DRQN |
|---|---|---|
| Frequency of learning | 5 | 3 |
| Memory size | 20000 | 20000 |
| batch size | 32 | 8 |
| learning rate | 0.0005 | 0.00006 |
| memory warmup size | 200 | 50 |
| gamma | 0.99 | 0.99 |

# Model hyperparameters:

Neural network view:

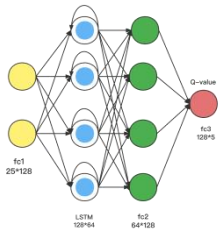
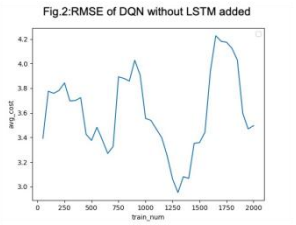


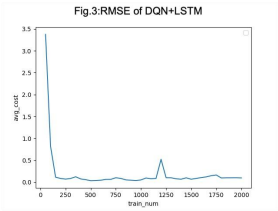
Fig.4 Neural network layers

Fig.4 is the specification of the neural network layers in this thesis. Since there is no picture, there is no convolutional neural network, it simply goes through the lstm network processing at the beginning before entering the hidden linear layer, which is a four-layer fully connected network.

**Evaluation Measures:**The RMSE was used for this evaluation and due to the large amount of data, we calculated the mean error for each episode.

# Results:






It can be seen that the DQN with the LSTM added converges very quickly to the error training, in marked contrast to the model without the LSTM added.

# Conclusion

The aim of this paper is to build recommenders that are more capable of capturing long-term user interest, an idea inspired by papers [2, 3, 4], where we added DQN to the LSTM and achieved good improvements, demonstrating that DRQN (Deep Recurrent Q Network) can not only achieve good results in games, but also for recommender systems with good It is shown that DRQN (Deep Recurrent Q Network) not only achieves good results in games, but also has good prediction capability for recommender systems, and has good performance and convergence speed. The recogym simulation environment [1] is used to simulate the dynamic interaction of the recommender system (agent) with the user, rather than static modelling with historical user data. It is also more compliant with privacy and security requirements and more socially civilised. The trial-and-error model of Reinforcement Learning (RL) helps us to adjust the optimal recommendation strategy, which is certainly different from traditional recommendations, and uses user feedback to improve the recommender accuracy in a simulated environment to recommend items that are more interesting and useful to the user.

# References

[1] Rohde, David & Bonner, Stephen & Dunlop, Travis & Vasile, Flavian & Karatzoglou, Alexandros. (2018). RecoGym: A Reinforcement Learning Environment for the problem of Product Recommendation in Online Advertising.

[2] Yan, A., Cheng, S., Kang, WC., Wan, M. & McAuley, J.J. (2019). CosRec: 2D Convolutional Neural Networks for Sequential Recommendation. Proceedings of the 28th ACM International Conference on Information and Knowledge Management.

[3] Elena Smirnova and Flavian Vasile. 2017. Contextual Sequence Modeling for Recommendation with Recurrent Neural Networks.

[4] Hausknecht, Matthew & Stone, Peter. (2015). Deep Recurrent Q-Learning for Partially Observable MDPs.

[5] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized Markov chains for next-basket recommendation. In Proceedings of the 19th international conference on World wide web (WWW'10). Association for Computing Machinery, New York, NY, USA, 811–820.