JDRec: Practical Actor-Critic Framework for Online Combinatorial Recommender System

Extended Abstract

Xin Zhao*
Tsinghua University
Beijing, China
zhaoxin19@gmail.com

Jiaxin Li*
Tsinghua University
Beijing, China
lijx23@mails.tsinghua.edu.cn

Zhiwei Fang
JD.com
Beijing, China
fangzhiwei2@jd.com

Yuchen Guo
Tsinghua University
Beijing, China
yuchen.w.guo@gmail.com

Jinyuan Zhao
JD.com
Beijing, China
zhaojy0618@126.com

Jie He
JD.com
Beijing, China
hejie67@jd.com

Wenlong Chen
JD.com
Beijing, China
chenwenlong17@jd.com

Changping Peng
JD.com
Beijing, China
pengchangping@jd.com

Guiguang Ding
Tsinghua University
Beijing, China
dinggg@tsinghua.edu.cn

ABSTRACT

In the realm of online recommendation systems, the Combinatorial Recommender (CR) system stands out for its unique approach. It presents users with a list of items on a result page, where user behavior is simultaneously influenced by contextual information and the items listed. Formulated as a combinatorial optimization problem, the objective of the CR system is to maximize the recommendation reward across the entire list of items. Despite the significant potential of CR systems, developing a practical and efficient model remains substantial challenges. These challenges stem from the dynamic nature of online environments and the pressing need for personalized recommendations. To tackle these challenges, we decompose the overarching problem into two sub-problems: list generation and list evaluation. We propose novel and pragmatic model architectures for each sub-problem aiming to concurrently enhance both effectiveness and efficiency. To further adapt the CR system to online scenarios, we integrate a bootstrap algorithm into an actor-critic reinforcement framework. This innovative approach called JD Recommender System (JDRec) is designed to continuously refine the recommendation mode through sustained user interaction, ensuring the system's adaptability and relevance. The proposed JDRec framework, tested through rigorous offline and online experiments, has shown promising results. It has been successfully deployed in online JD recommendation systems, yielding a notable improvement in click-through rate by 2.6% and augmenting the total value of the platform by 5.03%. Besides, we release the large scale dataset used in our work to facilitate further research.

*Equal contribution.

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand. © 2024 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

KEYWORDS

Recommender System; Combinatorial Recommendation; Reinforcement Learning

ACM Reference Format:
Xin Zhao, Jiaxin Li[1], Zhiwei Fang, Yuchen Guo, Jinyuan Zhao, Jie He, Wenlong Chen, Changping Peng, and Guiguang Ding. 2024. JDRec: Practical Actor-Critic Framework for Online Combinatorial Recommender System: Extended Abstract. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 3 pages.

1 INTRODUCTION

Recommender systems play a core role in e-commerce by effectively mining user preferences, whose core task is ranking the relevant items and presenting the result to users. Learning-to-rank refers to a series of methods that attempt to solve ranking problems with machine learning algorithms. Based on different designs of model formulation and loss functions, existing learning-to-rank algorithms can be categorized into four groups: point-wise [6], pair-wise [4], list-wise [2, 3], set-wise [7]. Most of the learning-to-rank methods are based on the probability ranking principle with an assumption that an item's attractiveness to a user remains constant regardless of what surrounding items are. This kind of method ignores inter-item relationship and local context information which are proved to be important for understanding user behaviors in online systems [5, 11].

There are two main context-aware ranking approaches: point-wise methods enhanced with contextual information, list-wise optimization via slate reranking. Contextual point-wise methods, like Deep Determinantal Point Process [13] and the Deep List-Wise Context Model [1], incorporate local contextual data and generate item lists using greedy algorithms. This kind of method takes the diversity of the lists into accounts, but ignores other user experience factors (e.g. category importance and correlation). On the other hand, slate reranking methods [8, 10, 12] utilize global list-wise

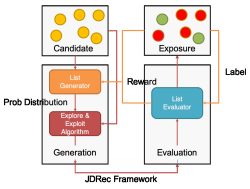

Figure 1: JDRec framework

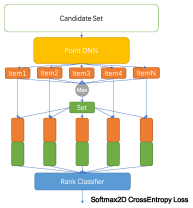

Figure 2: The list generator

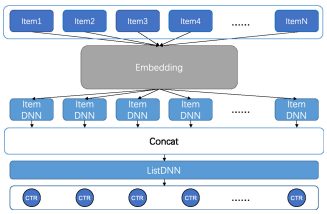

Figure 3: The list evaluator

optimization for better recommendations. Slate reranking methods try to present combinatorial recommendation from a global perspective instead of point-wise value estimation and sorting aloing with rule-based diversity control. There remains two challenges in slate reranking methods: how to generate superior candidate lists, and how to evaluate these lists to select the best one.

The principle of reinforcement learning - try, evaluation, feedback and improvement - is ideal for solving the slate reranking problem. The generator-critic framework belonging to the actor-critic reinforcement method [9] is proposed in industrial recommender system to solve the slate reranking problem. These approaches align with the actor-critic reinforcement method and seek to optimize both the list generation and the list evaluation. They propose RNN-based generators and attention-based evaluators. While transformative, this generator-critic framework presents the following challenges: (1) high time cost of RNN-based generators. (2) prediction bias of the list evaluator. (3) initiating the list generator before deploying it online presents a significant challenge.

In this work, we propose JDRec framework with a novel model structure for the list generator to reduce the cost of time online, and use a bootstrap approach to solve the initialization problem.

2 METHODOLOGY

The combinatorial recommendation problem can be formulated as follows: given a set of candidate items denoted as $I = \{I_1, I_2, ..., I_n\}$ and a specific user profile denoted as $\mathcal{U}$, the objective is to find an optimal sequence$\{\mathcal{A}\} = \{I_{a_1}, I_{a_2}, ..., I_{a_l}\}$, where $I_{a_i} \in \mathcal{I} (1 \leq i \leq l)$. This optimized sequence $\{\mathcal{A}\}$ is then presented to the user. Subsequently, the user provides feedback, represented as $r(\mathcal{U}, \mathcal{A})$. The objective is to maximize the expected overall utility or benefit of sequence $\mathcal{A}$, denoted as $\mathrm{E}\,[r(\mathcal{U}, \mathcal{A})]$. Typically, $\mathrm{E}_X[]$ represents the expectation over variable X, and $\mathrm{E}[]$ represents the expectation over repeated experiments.

JDRec framework (shown in Figure 1) is a variation of the actor-critic reinforcement learning framework tailored for recommender systems, which consists of a list generator and a list evaluator.

Table 1: An example of the list generator's output (L=4, N=6)

	1	2	3	4	5	6
1	0.9	0.1	0	0	0	0
2	0.05	0.8	0.15	0	0	0
3	0.05	0.1	0.7	0.15	0	0
4	0	0	0.15	0.85	0	0
Not in	0	0	0	0	1	1

The goal of the list generator is to model a mapping from candidate set to a list generation policy probability matrix. The structure of the proposed set-to-list generator is shown in Figure 2. The existing generator-critic methods tend to model list generation procedure as a recurrent generation paradigm, which has a high time cost online. Our proposed list generator takes as input a candidate item set, which is permutation invariant. The target of the list generator is to generate a probability distribution in the list generation procedure, which will be then used to sample several candidate item lists. The output of the set-to-list list generator is a $(L + 1) \times N$ 2D matrix M, where N represents the size of candidate set and L represents the length of the generated candidate list. Table 1 shows an example of the list generator's output. We propose the Softmax2D cross entropy loss for the list generator which is actually a mixture of two sub-tasks, comprising of id selection for each position and rank classification for each candidate item.

The goal of the list evaluator is to predict click-through rate for each item in the list. The model structure of the proposed list evalutor is shown in Figure 3. The list evaluator takes as input a sequence of items with user interactive features in user sessions, point-wise prediction results and other additional information of items. The output of the list evaluator is the overall click-through rate of the list. The training procedure of the list evaluator can be formulated as follows. Given an item list $\{\mathcal{A}\} = \{I_{a_1}, I_{a_2}, \ldots, I_{a_l}\}$ and its exposure label $\{\mathcal{L}_e\} = \{L_{e,a_1}, L_{e,a_2}, \ldots, L_{e,a_l}\}$ and click label $\{\mathcal{L}_c\} = \{L_{c,a_1}, L_{c,a_2}, \ldots, L_{c,a_l}\}$, where $L_{e,a_i} \in \{0, 1\}$ and $L_{c,a_i} \in \{0, 1\}$ and 1 means exposure or click respectively. The objective of the list evaluator is to estimate CTR of the candidate lists. So we use the sigmoid cross entropy loss as the loss function and ignore the loss computation on those unexposed items.

3 EXPERIMENTAL RESULTS

Table 2: A/B Test Results

Gray Release	click-through rate	total value
Evaluator (21st May)	+2.16%	+3.68%
Generator (9th Dec)	+0.44%	+1.35%

We conducted online evaluation of our JDRec framework, including A/B tests for 2 main steps. The A/B test results are shown in Table 2, where total value means synthetical value brought by user clicks (e.g., order and income). As shown in Table 2, the JDRec framework brings immediate gains for online recommendation, ensuring a seamless release process for the proposed JDRec framework.

REFERENCES

[1] Qingyao Ai, Keping Bi, Jiafeng Guo, and W. Bruce Croft. 2018. Learning a Deep Listwise Context Model for Ranking Refinement. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, Kevyn Collins-Thompson, Qiaozhu Mei, Brian D. Davison, Yiqun Liu, and Emine Yilmaz (Eds.). ACM, 135–144.

[2] Sebastian Bruch, Shuguang Han, Michael Bendersky, and Marc Najork. 2020. A Stochastic Treatment of Learning to Rank Scoring Functions. In *WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020*, James Caverlee, Xia (Ben) Hu, Mounia Lalmas, and Wei Wang (Eds.). ACM, 61–69.

[3] Sebastian Bruch, Masrour Zoghi, Michael Bendersky, and Marc Najork. 2019. Revisiting Approximate Metric Optimization in the Age of Deep Neural Networks. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, Benjamin Piwowarski, Max Chevalier, Éric Gaussier, Yoelle Maarek, Jian-Yun Nie, and Falk Scholer (Eds.). ACM, 1241–1244.

[4] Christopher J. C. Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Gregory N. Hullender. 2005. Learning to rank using gradient descent. In *Machine Learning, Proceedings of the Twenty-Second International Conference (ICML 2005), Bonn, Germany, August 7-11, 2005 (ACM International Conference Proceeding Series, Vol. 119)*, Luc De Raedt and Stefan Wrobel (Eds.). ACM, 89–96.

[5] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. 2016. Wide & Deep Learning for Recommender Systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems, DLRS@RecSys 2016, Boston, MA, USA, September 15, 2016*, Alexandros Karatzoglou, Balázs Hidasi, Domonkos Tikk, Oren Sar Shalom, Haggai Roitman, Bracha Shapira, and Lior Rokach (Eds.). ACM, 7–10.

[6] Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics* (2001), 1189–1232.

[7] Liang Pang, Jun Xu, Qingyao Ai, Yanyan Lan, Xueqi Cheng, and Jirong Wen. 2020. SetRank: Learning a Permutation-Invariant Ranking Model for Information Retrieval. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, Jimmy Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu (Eds.). ACM, 499–508.

[8] Changhua Pei, Yi Zhang, Yongfeng Zhang, Fei Sun, Xiao Lin, Hanxiao Sun, Jian Wu, Peng Jiang, Junfeng Ge, Wenwu Ou, and Dan Pei. 2019. Personalized re-ranking for recommendation. In *Proceedings of the 13th ACM Conference on Recommender Systems, RecSys 2019, Copenhagen, Denmark, September 16-20, 2019*, Toine Bogers, Alan Said, Peter Brusilovsky, and Domonkos Tikk (Eds.).

[9] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin A. Riedmiller. 2014. Deterministic Policy Gradient Algorithms. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014 (JMLR Workshop and Conference Proceedings, Vol. 32)*. JMLR.org, 387–395.

[10] Fan Wang, Xiaomin Fang, Lihang Liu, Yaxue Chen, Jiucheng Tao, Zhiming Peng, Cihang Jin, and Hao Tian. 2019. Sequential Evaluation and Generation Framework for Combinatorial Recommender System. *CoRR* abs/1902.00245 (2019).

[11] Ruoxi Wang, Bin Fu, Gang Fu, and Mingliang Wang. 2017. Deep & Cross Network for Ad Click Predictions. In *Proceedings of the ADKDD'17, Halifax, NS, Canada, August 13 - 17, 2017*. ACM, 12:1–12:7.

[12] Jianxiong Wei, Anxiang Zeng, Yueqiu Wu, Peng Guo, Qingsong Hua, and Qingpeng Cai. 2020. Generator and Critic: A Deep Reinforcement Learning Approach for Slate Re-ranking in E-commerce. *CoRR* abs/2005.12206 (2020).

[13] Mark Wilhelm, Ajith Ramanathan, Alexander Bonomo, Sagar Jain, Ed H. Chi, and Jennifer Gillenwater. 2018. Practical Diversified Recommendations on YouTube with Determinantal Point Processes. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy*, Alfredo Cuzzocrea, James Allan, Norman W. Paton, Divesh Srivastava, Rakesh Agrawal, Andrei Z. Broder, Mohammed J. Zaki, K. Selçuk Candan, Alexandros Labrinidis, Assaf Schuster, and Haixun Wang (Eds.). ACM, 2165–2173.