Imitate The World: A Search Engine Simulation Platform

Yongqing Gao∗
gaoyongqing2@smail.nju.edu.cn
Nanjing University
China

Guangda Huzhang∗
guangda.hzgd@alibaba-inc.com
Alibaba Group
China

Weijie Shen
shenweijie95@smail.nju.edu.cn
Nanjing University
China

Yawen Liu
liuyw@smail.nju.edu.cn
Nanjing University
China

Wen-Ji Zhou
eric.zwj@alibaba-inc.com
Alibaba Group
China

Qing Da
daqing.dq@alibaba-inc.com
Alibaba Group
China

Yang Yu
yuy@nju.edu.cn
Nanjing University
China

ABSTRACT
Recent E-commerce applications benefit from the growth of deep learning techniques. However, we notice that many works attempt to maximize business objectives by closely matching offline labels which follow the supervised learning paradigm. This results in models obtain high offline performance in terms of Area Under Curve (AUC) and Normalized Discounted Cumulative Gain (NDCG), but cannot consistently increase the revenue metrics such as purchases amount of users. Towards the issues, we build a simulated search engine AESim that can properly give feedback by a well-trained discriminator for generated pages, as a dynamic dataset. Different from previous simulation platforms which lose connection with the real world, ours depends on the real data in AliExpress Search: we use adversarial learning to generate virtual users and use Generative Adversarial Imitation Learning (GAIL) to capture behavior patterns of users. Our experiments also show AESim can better reflect the online performance of ranking models than classic ranking metrics, implying AESim can play a surrogate of AliExpress Search and evaluate models without going online.

CCS CONCEPTS
• Applied computing → Online shopping; • Computing methodologies → Simulation evaluation, Machine learning.

KEYWORDS
Learning-To-Rank, Simulation Evaluation, Dynamic Dataset

∗Both authors contributed equally to the paper.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Woodstock ’18, June 03–05, 2018, Woodstock, NY.
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-XXXX-X/18/06 . $15.00
https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

With increasing research activities in deep learning theory and practices, Learning-to-Rank (LTR) solutions rapidly evolve in many real-world applications. As the main component of an online system in E-commerce, LTR models strongly connect to the business profit. However, industrial LTR studies meet two stubborn issues. First, as the recent RecSys work [8] reveals, many works cannot be perfectly reproduced. Second, even we successfully reproduce the performances of proposed methods in a specified task, it is hard to promise the same performance in another task, as well as in the online environments. Therefore, it is desired for researchers to have a public platform for E-commerce LTR evaluation.

A typical process of industrial search engines contains three stages to produce a display list from a user query. A search engine first retrieves related items with intend of the user (i.e. the user query), then the ranker ranks these items by a fine-tuned deep LTR model, finally, the re-ranker rearranges the order of items to achieve some businesses goals such as diversity and advertising. Our proposed simulation platform AESim contains these three stages. We replace queries with category indices in our work, therefore AESim can retrieve items from a desensitized items database by the
category index. After that, a customizable ranker and a customizable re-ranker produce the final item list. AESim allows us to study joint learning of multiple models, we left it as future work and focus on the correct evaluation for a single model.

Besides the set of real items, two important modules make AESim vividly reflect the behaviors of real users. Virtual user module aims at generating embeddings of virtual users and their query, and it follows the paradigm of Wasserstein Generative Adversarial Network with Gradient Penalty (WGAN-GP). Feedback module inputs the display list and the information of the user, then outputs the feedback of users on the display list. To model the decision process of users, we train the feedback module by Generative Adversarial Imitation Learning (GAIL). For diversifying behaviors, we consider clicking and purchasing, which are two of the most important feedback of users in E-commerce.

The contribution of AESim includes:
- As far as we know, AESim is the first E-commerce simulation platform generated by imitating real-world users.
- AESim can be used as a fair playground for future studies on E-commerce LTR researches.
- Our online A/B testings show that AESim can reflect online performance without online interaction.

2 RELATED WORKS

Generally, most Learning-to-rank (LTR) models are partitioned into three groups: point-wise models, pair-wise models, and list-wise models. These methods have different forms of loss functions. Point-wise models [7, 12, 17] focus on an individual classification or regression task. The loss of pair-wise models [4, 5, 15, 18, 20] include pairs of scored items, and it is computed by the relative relationship of their scores and labels. List-wise models [1, 6, 24, 25] score items to optimize the holistic metrics of lists. Practically, all these models give item scores and the online system will rank items straightforwardly by the scores. However, evaluating models by historical data is problematic, which may lead the online-offline inconsistency [3, 13, 19, 21, 22].

To correctly evaluate a model without going online, a simulation platform is necessary to give a dynamic response for a newly generated list. There are several simulation platform for search engines and recommender systems, such as Virtual-Taobao [23], RecSim [14] and RecoGym [21]. However, Virtual-Taobao cannot give an evaluation for a complete list. RecSim and RecoGym can evaluate reinforcement learning models, but they lose the connection to real-world application. Our model follows generative adversarial imitation learning (GAIL) [10], which has been examined to be a better choice for imitation learning [9, 11, 23], to learn the patterns of real users.

3 THE PROPOSED FRAMEWORK

AESim includes an item database, a virtual user module, a feedback module, a customizable ranker system, and generated datasets. It can test LTR algorithms with a straightforward evaluation and can test de-biasing methods in a pure offline environment. The item database contains millions of selected active items and these items are categorized with their category indices. To train and evaluate a ranker model, AESim first prepares the training set and the testing set of labeled lists by the virtual user module (generate queries), a complete ranker system (generate final lists), the feedback module (generate feedback of virtual users). With the training set, we can train new ranker models and produce results for the testing set. Finally, we use the feedback module again to examine the true performance of the ranker model.

3.1 Virtual User Module

Virtual User Module contains a generator and a discriminator which are trained following WGAN-GP. The generator aims at generating features of users and his query which are similar to the real records. The discriminator tries to distinguish the fake/generated and real pairs of users and queries and guides the generator to reach its
3.2 Ranker System of AESim

The process of ranking in AESim is similar to real search engines. After the virtual user module generates a user-query pair, the ranker system inside AESim starts to compute the final display list. First, it retrieves 1000 items from the item dataset with the query, which is translated into the category index in our work. Then, the ranker (a point-wise model) scores the items and sends the top 50 of them to the re-ranker, and the re-ranker decides the final order of items. Finally, AESim evaluates the output of the ranker system by the feedback module.

3.3 Feedback Module

Our feedback module has a classic sequence-to-sequence structure and rewards each item by imitating the behaviors of real users. We train its parameters follows GAIL: a discriminator in included to distinguish how the generated behaviors are close to the behaviors of real users. In our work, we also try to use WGAN-GP to generate feedback. The outputs of the feedback module following GAIL are much more similar to real behaviors than the one following WGAN-GP. In GAIL, the gradient of discriminator has the following form:

$$\nabla_{\theta_D} \log D(s, a|\theta_D)) + \beta r_s \left[ \nabla_{\theta_G} \log (1 - D(s, a|\theta_D)) \right]$$

Here $$\theta_D$$ is the generated trajectory with parameters $$\theta_G$$ and $$r_s$$ is the real trajectory. State s and action a are included in the trajectories. The parameters of generator $$\theta_G$$ is updated with reward function $$\log (D(s, a|\theta_D))$$ using the TRPO rule.

Compared to previous simulation platforms, ours has a significantly similar purchase trend to real-world applications. Figure 5 shows the conversion rate of the purchase at each position with the log function. The trajectories. The parameters of generator $$\theta_G$$ is updated with reward function $$\log (D(s, a|\theta_D))$$ using the TRPO rule.

3.4 Dataset Preparation

To build the dataset for model training and testing, we further need rankers which help the ranker system produce the initial dataset. We first use a random weight ranker to generate a random training set for training a base ranker. Then we use a base ranker to generate the final training dataset and testing dataset for model evaluation. An important benefit of the above steps is that we can reproduce the sample selection bias issue of offline data.

The training set in AESim is the same as the traditional static dataset for supervised learning models. The main difference appears

Figure 3: The simulation effect of AESim.
De-biasing

Table 1: The model results in AESim.

| Method   | GAUC   | NDCG   | MAP    | AESim   | GAUC   | NDCG   | MAP    | AESim   |
|----------|--------|--------|--------|---------|--------|--------|--------|---------|
| Point-wise | 0.806283 | 0.623264 | 0.025052 | 0.003089 | 0.805345 | 0.620394 | 0.024876 | 0.003074 |
| Pair-wise | 0.805478 | 0.621492 | 0.024959 | 0.003087 | 0.804045 | 0.618290 | 0.024746 | 0.003084 |
| ListMLE  | 0.799506 | 0.626811 | 0.025345 | 0.002984 | 0.794266 | 0.629366 | 0.025500 | 0.002952 |
| Group-wise | 0.806052 | 0.634064 | 0.025770 | 0.002657 | 0.807156 | 0.633718 | 0.025757 | 0.002615 |
| DLCM     | 0.807749 | 0.634064 | 0.025770 | 0.002657 | 0.807156 | 0.633718 | 0.025757 | 0.002615 |

Table 2: Online performance and AESim evaluations of models.

|        | AESim | Day 1 | Day 2 | Day 3 | Day 4 | Day 5 | Day 6 | Day 7 | Day 8 | Day 9 | Day 10 |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| Point-wise | +0.00%  | +0.54%  | +0.59%  | +1.11%  | +0.70%  | +0.62%  | +1.15%  | +0.29%  | -0.24%  | +0.12%  | +0.31%  |
| Pair-wise  | -0.26%  | -0.39%  | -0.05%  | +1.02%  | +0.25%  | +0.39%  | +1.34%  | +0.86%  | -0.60%  | -0.19%  | -0.85%  |
| ListMLE    | -1.00%  | -2.15%  | -1.51%  | -0.29%  | -1.46%  | -2.57%  | -2.17%  | -2.24%  | -2.83%  | -2.32%  | -3.81%  |
| DLCM       | -17.1%  | -1.34%  | -0.42%  | +0.19%  | -1.52%  | -2.72%  | -1.39%  | -1.95%  | -0.31%  | -0.48%  | -1.36%  |

Figure 5: The purchase trend of users at each position in real and fake scenarios.

4 EXPERIMENT

Offline Testing. We test a point-wise method with a cross-entropy loss, a pair-wise method with a logistic loss [5], the listMLE [1], and the group-wise scoring framework [2] (GSF) in AESim, where all these methods use the same MLP (note that GSF contains several isomorphic MLP). We further add DLCM which is expected to have a high offline performance for its complicated structure.

To include the de-biasing methods, we proceed with a simulation in AESim to swap the first item and the $k$-th items, then observe the change of conversion rate to determine the value of position bias [16]. After that, some of the above methods can add an inverse propensity score to remove the influence brought by position bias. It can be observed in Table 1 that GAUC, NDCG and MAP have similar preferences for models but AESim scores give different orders. Especially, DLCM gets the highest GAUC but obtains low AESim scores, which implies that a model with high GAUC may fail to optimize the online performance.

Online Testing. To examine that AESim correctly evaluates the models, we put the point-wise model, the pair-wise model, and the ListMLE in our online system. Each model needs to serve a non-overlapping random portion of search queries as a re-ranker. Roughly, each model serves millions of users and produces millions of lists per day. Due to the daily dramatic change of online environments, the difference gap may perform unstably so that we need to consider the overall performance of models. The ten days result in Table 2 shows the consistency with our offline evaluation for the point-wise model, the pair-wise model, and the ListMLE. However, DLCM is evaluated extremely poor in AESim and its performance is not that bad when serves online. Therefore, we suggest considering AESim as a rough judgment for a model, which may have a gap with the actual performance.

5 CONCLUSION

We propose an E-Commerce search engine simulation platform for model examinations, which was a missing piece to connect evaluations of LTR researches and business objectives of real-world applications. AESim can examine models in the simulation E-commerce environment with dynamic responses, and its framework can be easily extended to other scenarios that items and users have different features. We hope to see the development of a dynamic dataset that facilitates industrial LTR researches in the future.
