KuaiSim: A Comprehensive Simulator for Recommender Systems

Appendix

A Detailed Dataset Description and Analysis

A.1 Detailed Feature Description

Table 1: Detailed interaction information in KuaiRand dataset.

Interaction field	Feature type	Explanation
user id	int64	The unique identifier for the user.
video id	int64	The unique identifier for the video.
date	int64	The date when the interaction occurred.
hour min	int64	The time of the interaction in hours and minutes.
time ms	int64	The timestamp of the interaction in milliseconds.
is click	int64	The user feedback of click.
is like	int64	Indicating if the user liked the video.
is follow	int64	Indicating if the user followed the author.
is comment	int64	Indicating if the user wrote a comment.
is forward	int64	A binary signal indicating if the user forwarded the video.
is hate	int64	A binary signal indicating if the user disliked the video.
long view	int64	Indicating the completeness of the video.

Table 2: Rich user feature in KuaiRand dataset.

User feature	Feature type	Explanation
user id	int64	The unique identifier for the user.
user active degree	str	The level of user activity, classified as 'high active', 'full active', 'middle active', or 'UNKNOWN'.
is live streamer	int64	Indicates whether the user is a live streamer.
is video author	int64	Indicates if the user has uploaded any videos.
follow user num range	str	The number range of followed users.
fans user num range	str	The number range of fans.
friend user num range	str	The number range of friends.
register days range	str	The range of the number of days since user registration.
one-hot features	int64	Encrypted features of the user.

This section offers an elaborate overview of the KuaiRand [1] dataset, presenting its key components in detail. Because online A/B test usually consumes much time and money, which makes it impractical for academic researchers to conduct the evaluation online. However, offline evaluation will cause bias because of massive missing data, i.e., the user-item pairs that have not occurred in the test set. A way to fundamentally solve this problem in offline evaluation is to collect unbiased data, i.e., to elicit user preferences on the randomly exposed items. To achieve this goal, sample a batch of videos and filter out the spam such as advertisements. There are 7,583 items in total. For the target users, randomly select a batch of users and remove robots, which includes over 200,000 real users. Each time the recommender system recommends a video list to a user, decide whether to insert a random item with a fixed probability. If the answer is yes, then intervene in the recommendation list by randomly

Preprint. Under review.

Table 3: Rich video feature in KuaiRand dataset.

User feature	Feature type	Explanation
video id	int64	The unique identifier for the video.
author id	int64	The unique identifier for the author of the video.
video type	str	The type of the video, categorized as "NORMAL" or "AD".
upload type	str	The upload type of the video.
music type	int64	The background music type used in the video.
tag	str	A list of key categories or labels associated with the video.

selecting one video from this list and replacing it with a random item uniformly sampled from the 7,583 items. KuaiRand removes the users that have been exposed to less than 10 randomly exposed videos for faithful evaluation. There are 27,285 users retained. All 7,583 items have been inserted at least once, and the total number of random interventions is 1,186,059. The KuaiRand dataset encompasses five distinct components: an interaction dataset, a dataset comprising rich user features, and a dataset consisting of rich video features. We provide a detailed compilation of these components we used, including the names, feature types, and explanations, which are presented in Table 1 for interaction features, Table 2 for user features, and Table 3 for item features. We have meticulously selected high-quality features for both the user and item categories. From the 17 available one-hot features for the user, we have selected the following six features: [0, 1, 6, 9, 10, 11]. These specific features have been chosen to provide meaningful insights and additional dimensions for user analysis within the dataset. These features have been thoughtfully curated to ensure their relevance, reliability, and usefulness in gaining meaningful insights and understanding the user dynamics within the dataset. For detailed information about all the features included in the KuaiRand dataset, you can refer to the main page of the dataset [1].

A.2 Further data analysis

Figure 1 presents a heatmap that illustrates the association matrix of seven different types of feedback. The heatmap provides a visual representation of the relationships between these feedback categories. When users provide negative feedback, such as expressing hate towards a video, they tend to give very few other positive feedback signals. On the other hand, among the positive feedback signals, both clicks and long views exhibit a strong association with other positive feedback indicators. Conversely, forwarding a video shows the least association with other positive feedback signals. These observations highlight distinct patterns in user behavior when it comes to expressing negative feedback and engaging in various positive feedback actions.

Furthermore, Figure 2 provides valuable insights into the distribution of the user features utilized in our model. Regarding the feature 'user active degree', the majority of users are classified as 'full active', followed by 'high active', while the remaining categories represent a significantly smaller proportion. In terms of the 'live streamer' feature, the vast majority of users (-124) are not live streamers. However, when considering the 'video author' feature, a large majority of users are indeed video authors. Examining the 'follow user num range' feature, the category '500+' dominates the distribution, while the category '0' represents a minimal proportion. Analyzing the 'fans user num range' feature, the range '[10, 100]' captures the largest share, with both '[1, 10]' and '[100, 1k]' accounting for substantial percentages, and the remaining ranges having little to no representation. For the 'friend user num range' feature, the majority of users fall within the range '[5, 30]'. Lastly, the 'register days range' feature shows that the category '[730+]' encompasses nearly all users, indicating a significant proportion of long-standing registered accounts. This visual representation allows for a comprehensive understanding of the characteristics and patterns present within the user data.

Similarly, Figure 3 visually represents the distribution of video features employed in our model. Examining the 'video type' feature, the majority of videos fall under the category of 'NORMAL', while the 'UNKNOWN' category represents a significantly smaller proportion. Regarding the "music type" feature, the value '9.0' dominates the distribution, indicating a prevalent use of a specific background music type. Additionally, a substantial proportion of videos have no background music at all. Analyzing the 'upload type' feature, the category 'LongImport' captures the largest share, while the categories 'LipsSync' and 'PhotoCopy' account for a significantly smaller proportion. This visualization aids in the exploration of the video dataset, enabling the identification of noteworthy

[1] https://kuairand.com/

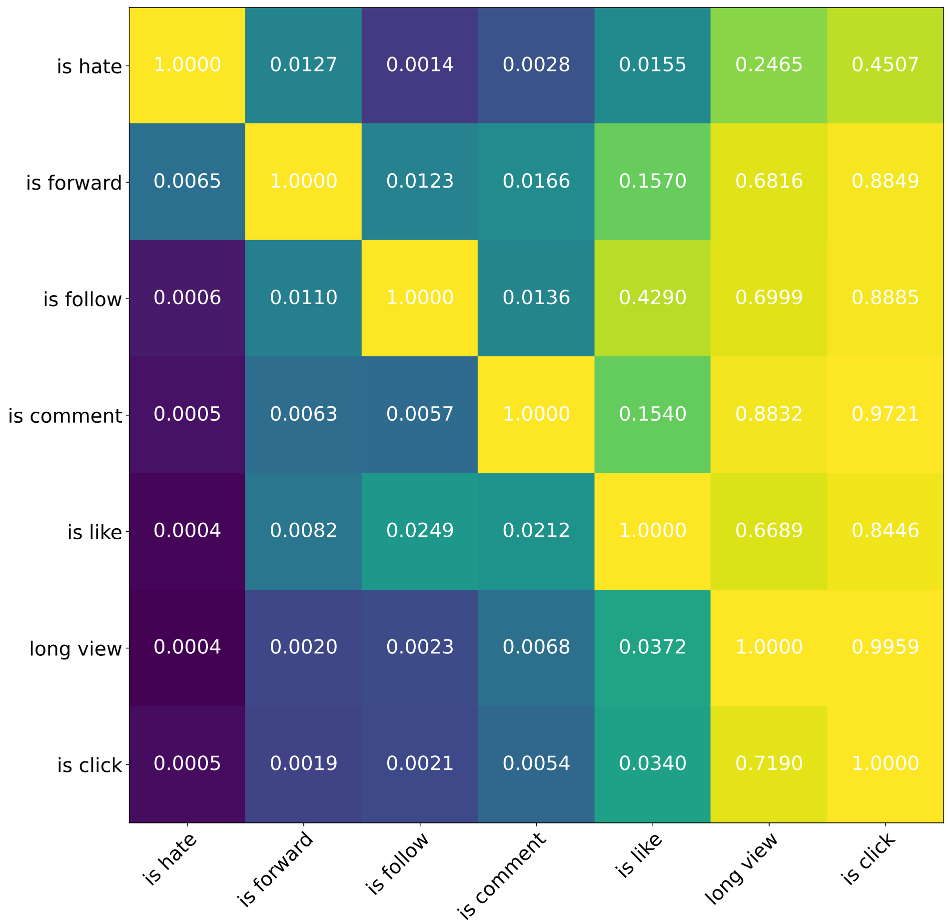

Figure 1: Association matrix of seven kinds of feedback.

trends or peculiarities within the data. These figures collectively enhance our understanding of the underlying patterns, associations, and distributions within the datasets, thereby facilitating a more robust and insightful model.

B Detailed Experiment Implementations

To ensure simplicity and consistency, we have established certain parameter settings and search spaces for our experiments. We set the embedding size of $\mathcal{U}$ and $\mathcal{H}_{:t-1}$ from [32, 64, 128]. The latent embedding size is two times the input dimension. In the immediate response module, we assign an immediate reward weight of 1 to all immediate feedback signals except for the hate signal, which is assigned a weight of -1. Consequently, the range of immediate reward, denoted as r, spans from -1 to 6 in the KuaiRand dataset, and from -1 to 2 in the ML-1m dataset. The layer number of DNN and Transformer is set to 2. The multi-head of the Transformer is set to 2. The dropout rate is set to 0.2. In the user leave module, we set the max time step and initial temper value from [5, 10, 15, 20, 25, 30], and the rate of temper decrease as 1. Additionally, we define the leave threshold as 1. Furthermore, in the user retention module, λ_1 is set to 0.5, and λ_2 is set to 0.75. The number of DNN layers is set to 2. The slate size is from [5, 10, 15, 20, 25, 30]. For the actor learning rate, we explore a common search space consisting of [0.0005, 0.0001, 0.00005, 0.00001, 0.000005, 0.000001]. Similarly, we investigate the critic learning rate in the range of [0.001, 0.0001, 0.00001]. For both simulator and agent training, the batch size is set to 64. And we search for the optimal learning rate within the range of [0.0005, 0.0001, 0.00005, 0.00001]. Moreover, we perform L2 regularization with coefficients selected from [0.0001, 0.00005, 0.00001, 0.000005]. When comparing to other baselines, we either utilize the same search range or adopt the optimal settings recommended by the original authors of the baselines. We divide the dataset as training set and test set with a ratio of 8:2, which is a common

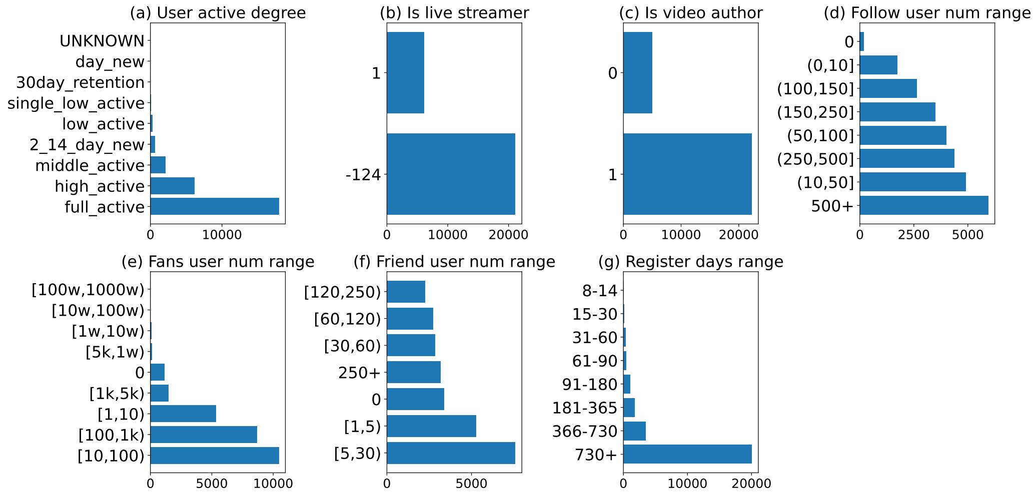

Figure 2: Rich user feature distribution. (a) User active degree. (b) Is live streamer. (c) Is video author. (d) Follow user num range. (e) Fans user num range. (f) Friend user num range. (g) Register days range.

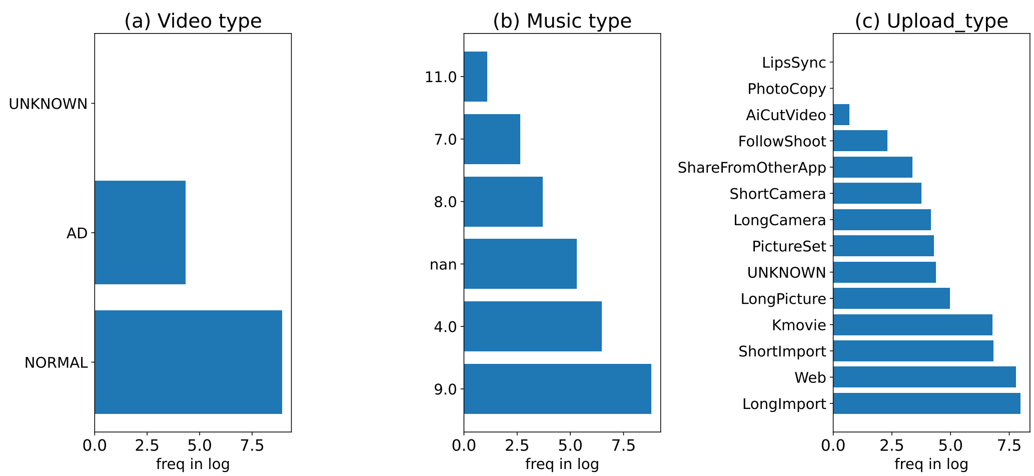

Figure 3: Rich video feature distribution. (a) Video type. (b) Music type. (c) Upload type.

setting in previous works. All experiments are conducted on an NVIDIA Tesla V100S GPU, and the reported results are the average of three replicate experiments.

Reproduction guidelines Below we will illustrate how to reproduce our experimental results. Please use Python 3.8 (or later), torch 2.0.0 (or later). If you want to use GPU acceleration, please use CUDA 11.8 (or later). Then you can follow the guidelines below:

- Install the necessary packages according to our '0.Setup'.
- Find the details of data preprocessing in 'code/preprocess/KuaiRandDataset.ipynb'.
- Run 'run_nultibehavior.sh' to train the simulator.
- Run 'generate_session_data.sh' to generate session data for agent training.
- Then you can run the 'train_(agent name)_krpure_(task level).sh' file to train different agents on different tasks with our simulator.
- We provide visualized training results in 'TrainingObservation.ipynb'.

We have provided detailed parameter settings in all the sh files. For more analysis experiments, you can also check out the project READ ME. And we will be working on further improvements to our project, such as readability guarantees for existing code and the incorporation of novel baselines.

References

[1] Chongming Gao, Shijun Li, Yuan Zhang, Jiawei Chen, Biao Li, Wenqiang Lei, Peng Jiang, and Xiangnan He. 2022. KuaiRand: An Unbiased Sequential Recommendation Dataset with Randomly Exposed Videos. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 3953–3957.