UserSim: User Simulation via Supervised Generative Adversarial Network

Xiangyu Zhao[1], Long Xia[2], Lixin Zou[3]

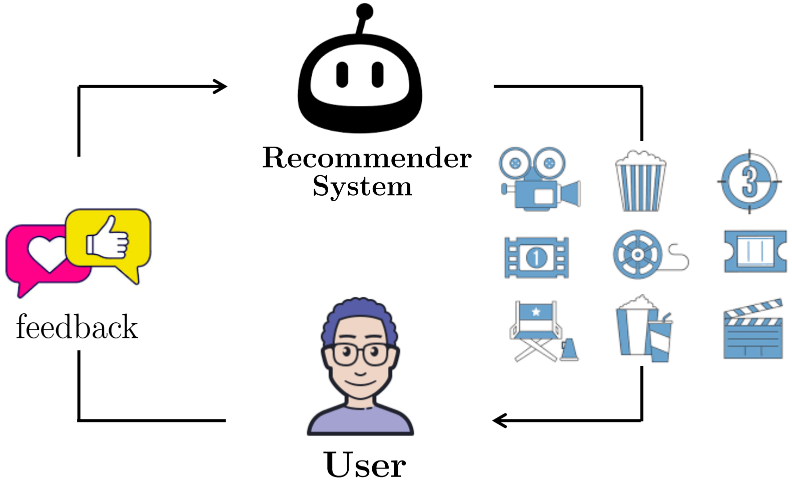
Hui Liu[1], Dawei Yin[3], Jiliang Tang[1]

1: Data Science and Engineering Lab, Michigan State University

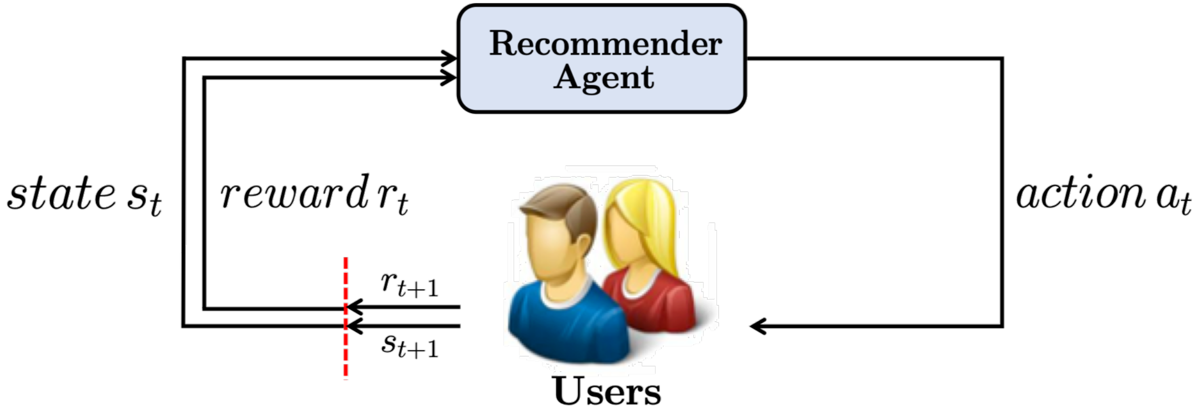
2: York University 3: Baidu

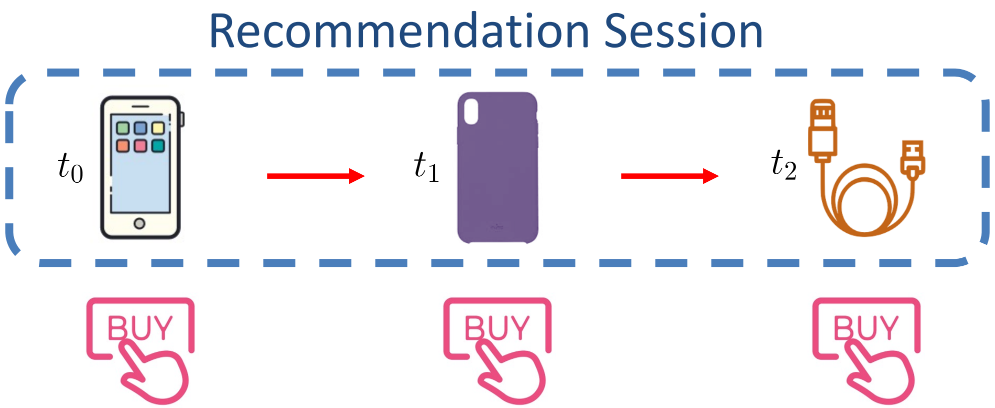
Reinforcement Learning for Recommendations

- Increasing interests in applying Reinforcement Learning for recommendations



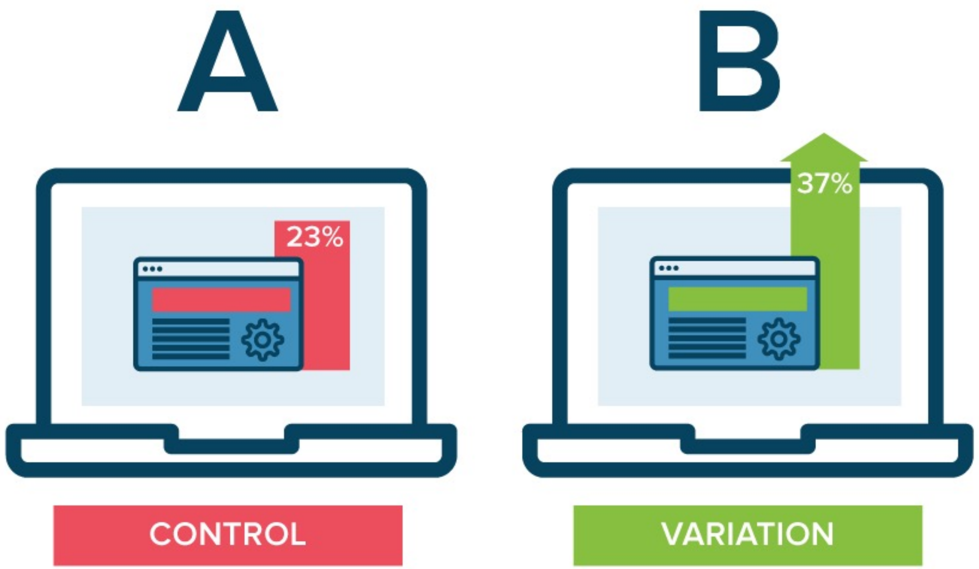
- Advantages
  - Continuously updating the recommendation strategies during the interactions
  - Maximizing the long-term reward from users



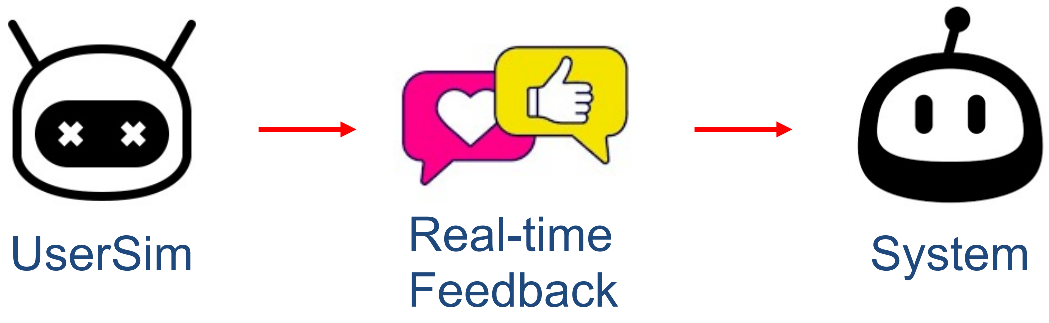


Real-time Feedback

- The most practical and precise way is online A/B test

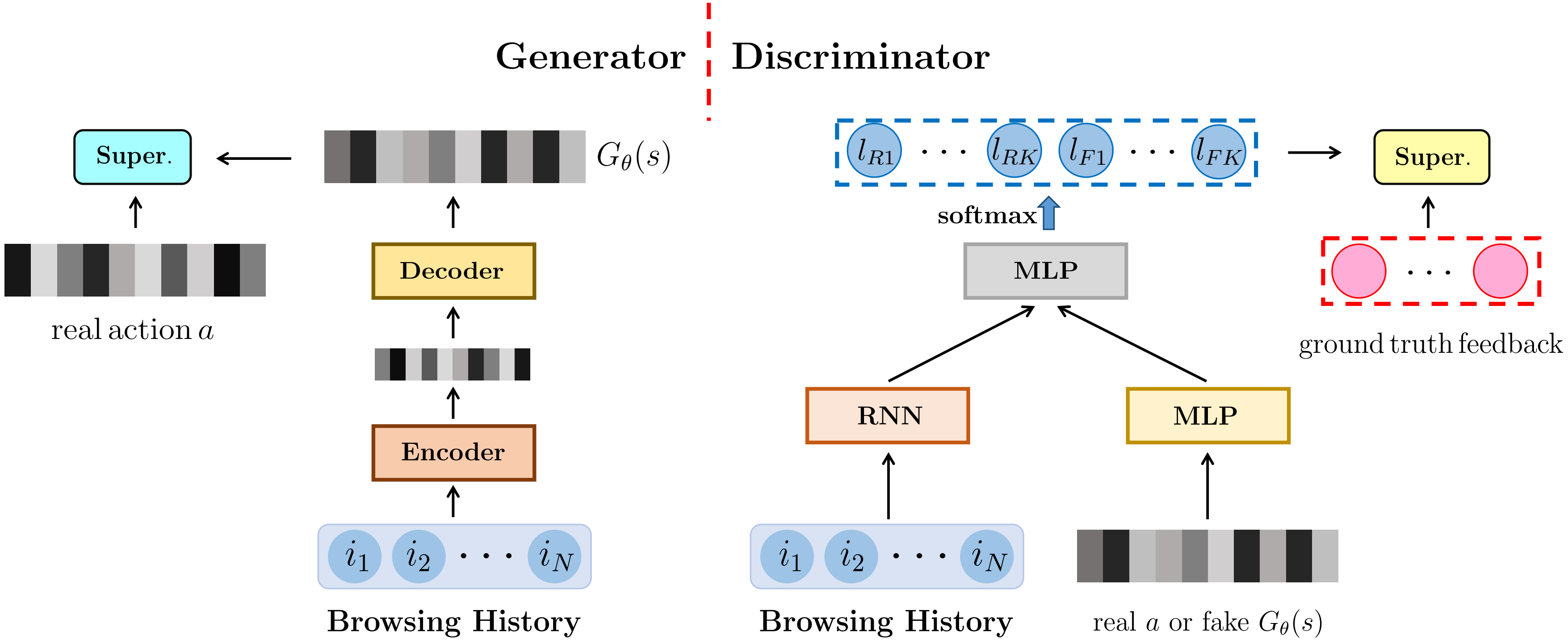


- Online A/B test is inefficient and expensive
  - Taking several weeks to collect sufficient data
  - Numerous engineering efforts
  - Bad user experience



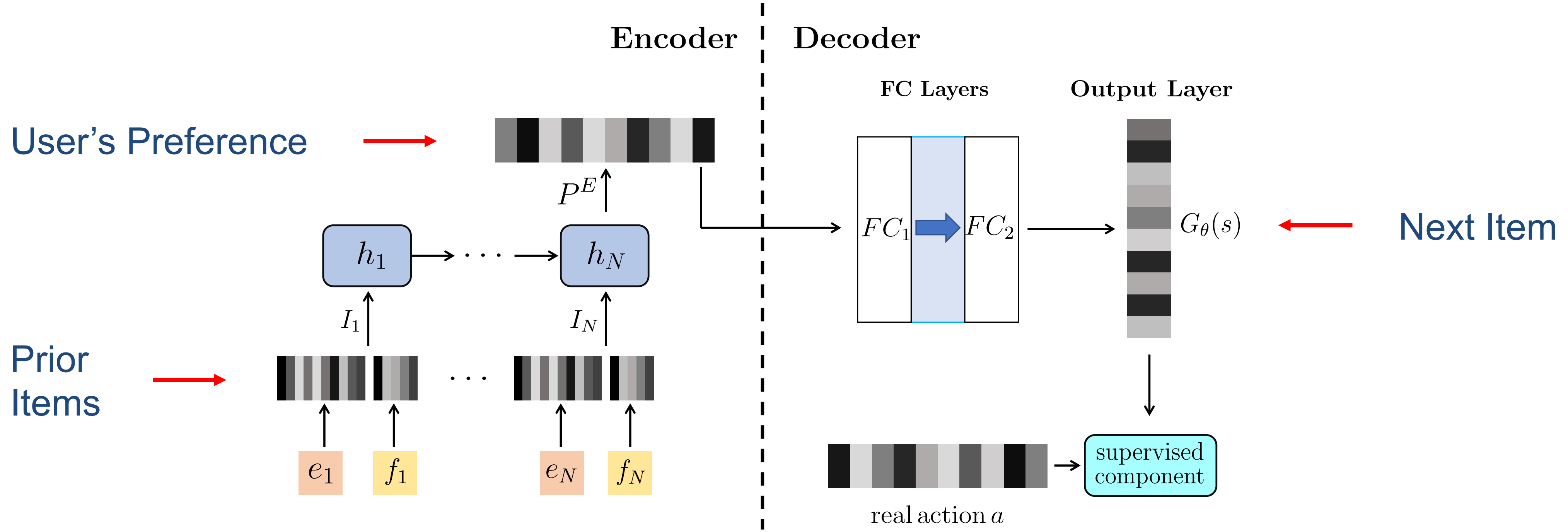
Overview

- Simulating users' real-time feedback is challenging
  - Underlying distribution of item sequences is extremely complex
  - Data available to each user is rather limited



Generator

- Learning the data distribution
- Generating indistinguishable logs based on users' browsing history



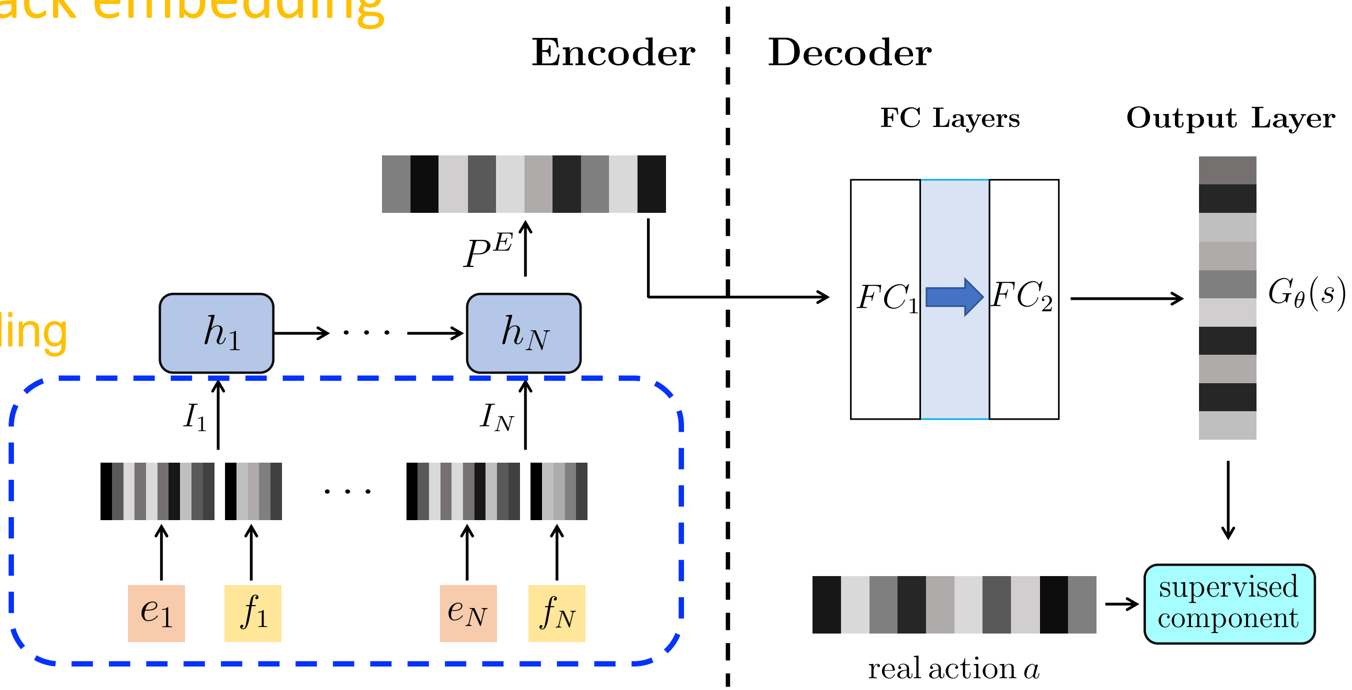
Encoder

- Input layer
  - e_n: item's identifier embedding
  - f_n: user's feedback embedding

Feedback Embedding

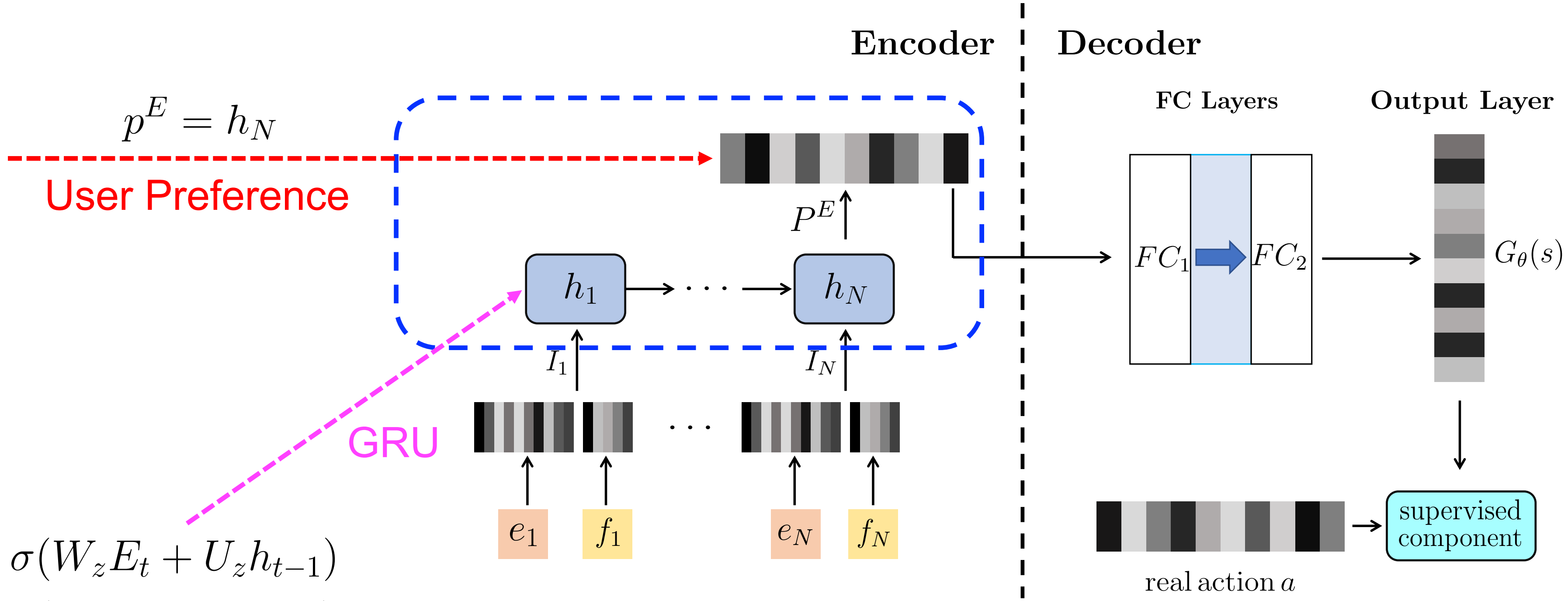
$$I_n = concat(e_n, f_n)$$

Identifier Embedding



Encoder

- GRU layer:
  - Capturing user's preference from the sequence of items



$$z_t = \sigma(W_z E_t + U_z h_{t-1})$$

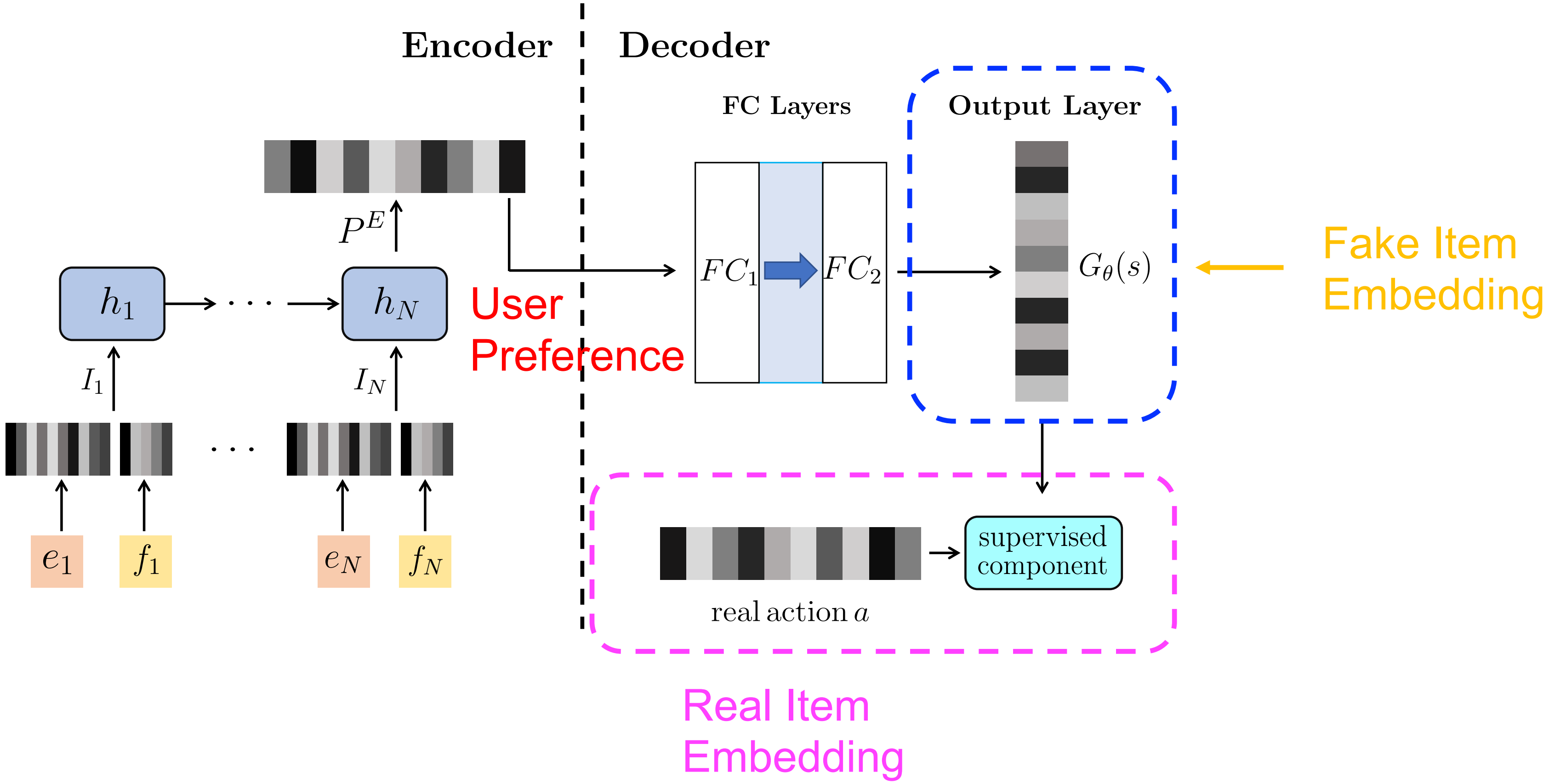
$$r_t = \sigma(W_r E_t + U_r h_{t-1})$$

$$h_t = (1 - z_t) h_{t-1} + z_t \hat{h}_t$$

$$\hat{h}_t = \tanh[W E_t + U(r_t \cdot h_{t-1})]$$

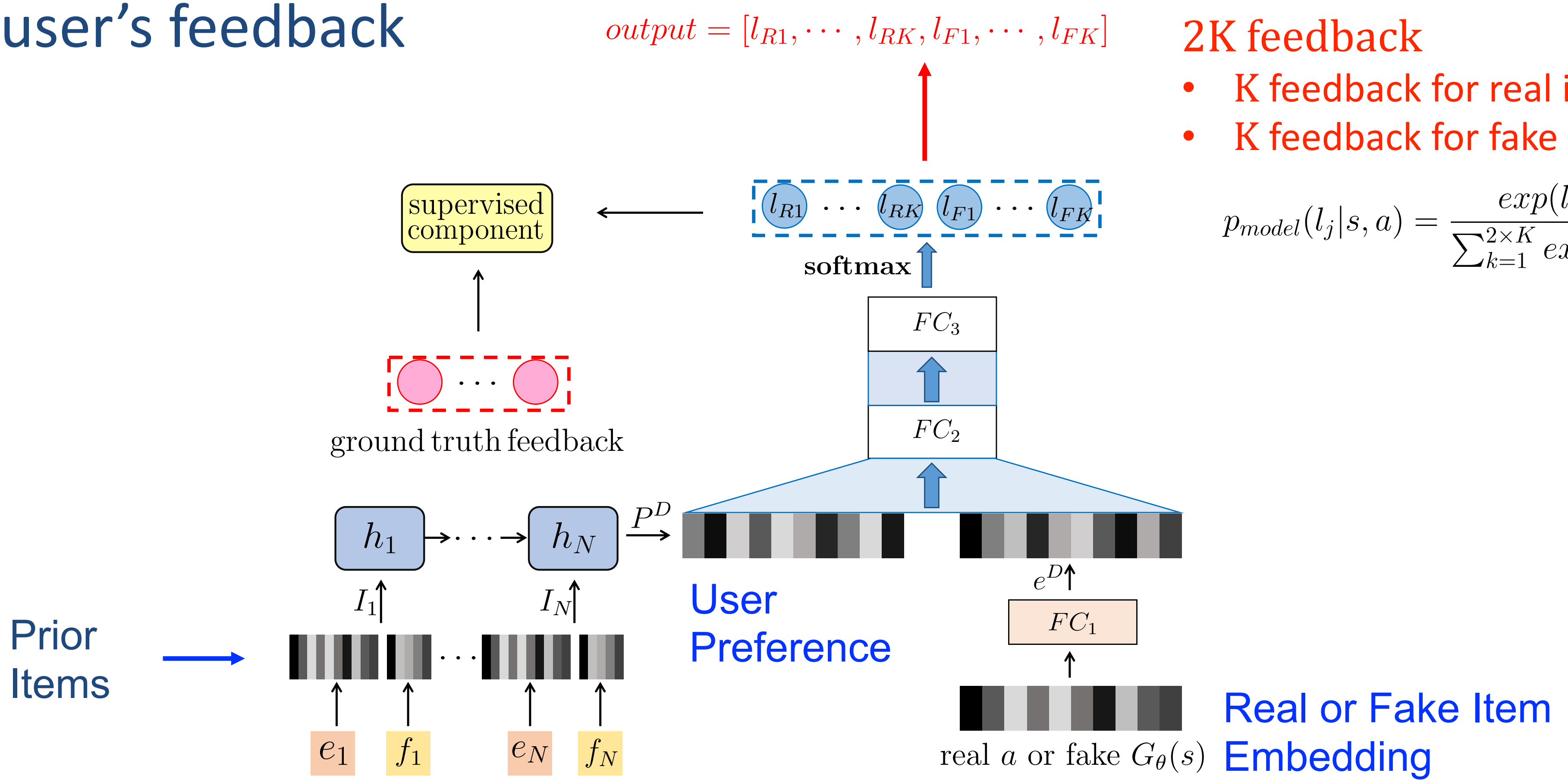
Decoder

- Goal:
  - Predicting the item to be recommended



Discriminator

- Distinguishing real/fake items
- Predicting user's feedback



2K feedback

- K feedback for real items
- K feedback for fake items

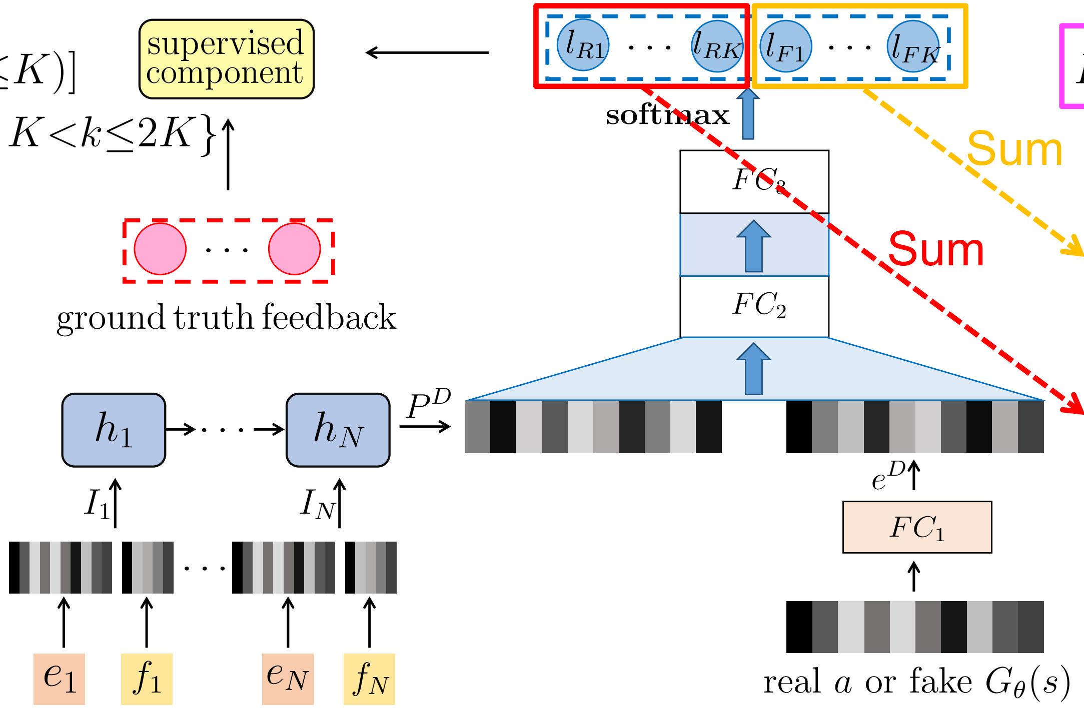
$$p_{model}(l_j|s,a) = \frac{exp(l_j)}{\sum_{k=1}^{2\times K} exp(l_k)}$$

Optimization

- Discriminator

$$L_D = \boxed{L_D^{unsup}} + \alpha \cdot \boxed{L_D^{sup}}$$

$$\boxed{L_D^{sup}} = -\{\mathbb{E}_{s,a,r\sim p_{data}}[\log p_{model}(l_k|s,a,k\leq K)] + \lambda \cdot \mathbb{E}_{s,r\sim p_{data}}[\log p_{model}(l_k|s,G_\theta(s),K<k\leq 2K\}$$



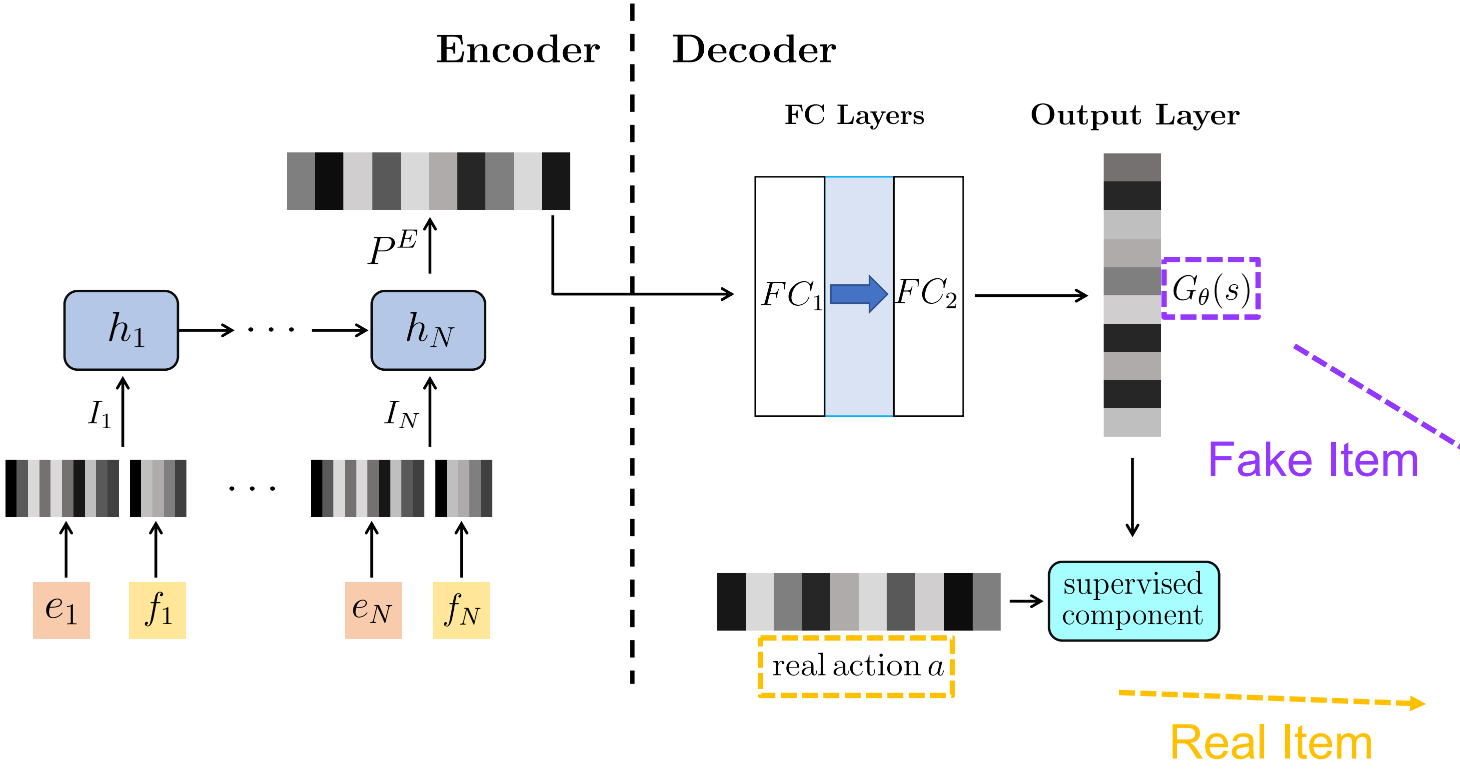
$$\boxed{L_D^{unsup}} = -\{\mathbb{E}_{s,a\sim p_{data}} \log D_\phi(s,a) + \mathbb{E}_{s\sim p_{data}} \log D_\phi(s,G_\theta(s))\}$$

$$D_\phi(s,G_\theta(s)) = \sum_{k=K+1}^{2\times K} p_{model}(l_k|s,G_\theta(s))$$

$$D_\phi(s,a) = \sum_{k=1}^{K} p_{model}(l_k|s,a)$$

Optimization

- Generator



$$\boxed{L_G^{unsup}} = \mathbb{E}_{s\sim p_{data}}[\log D_\phi(s, G_\theta(s))]$$

$$L_G = \boxed{L_G^{unsup}} + \beta \cdot \boxed{L_G^{sup}}$$

$$\boxed{L_G^{sup}} = \mathbb{E}_{s,a\sim p_{data}} \| \boxed{a} - \boxed{G_\theta(s)} \|_2^2$$

Experimental Settings

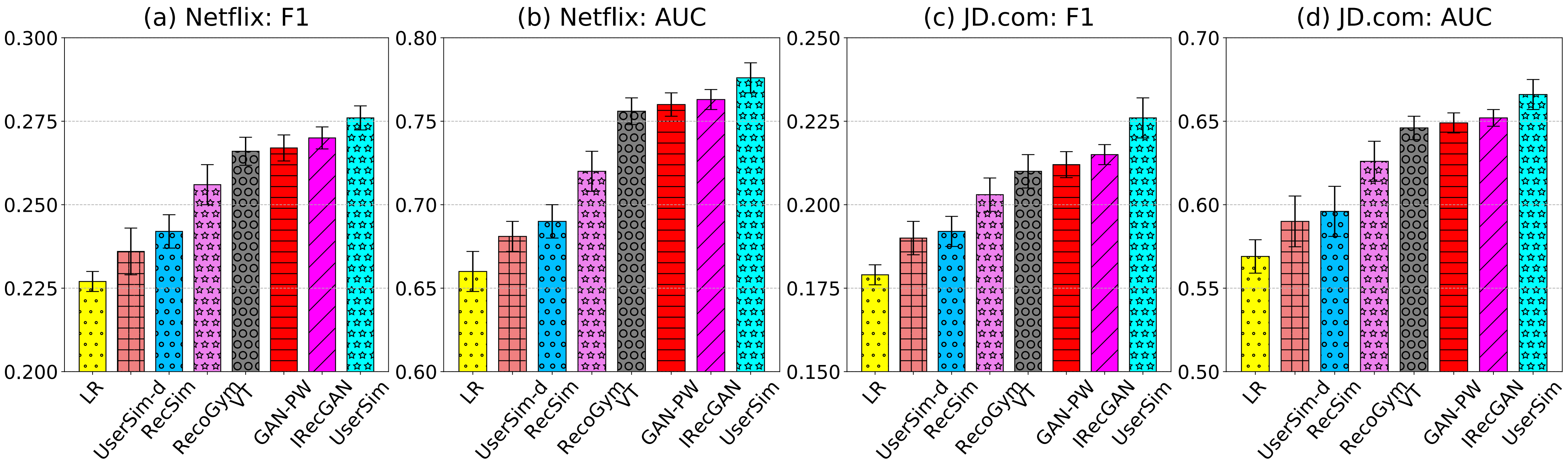
- Pubilc benchmark datasets
  - Netflix and JD.com
  - 70%: training/validation set
  - 30%: test set
- 4 types of feedback
  - Real-positive
  - Real-negative
  - Fake-positive
  - Fake-negative
  - Real: real item from data
  - Fake: fake item from generator

Object	Netflix Prize	JD.com
# user (session)	480,189	283,228
# item	17,770	1,355,255
# interaction	100,480,507	97,713,660
# ave. length	209	345
# feedback	rating 1~5	skip, click

4~5: positive
1~3: negative

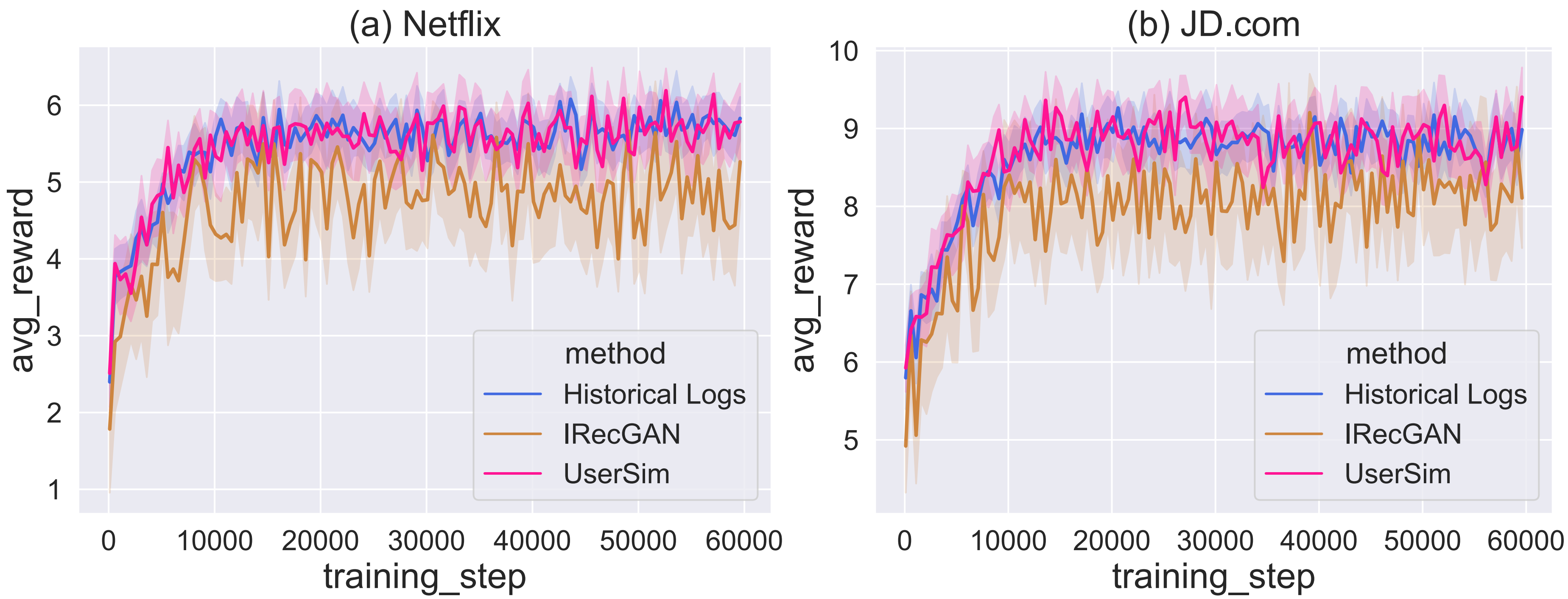
click: positive
skip: negative

Overall Performance



- Metric: F1-score
- Baselines: LR, UserSim-d, RecSim, RecoGym, Virtual-Taobao, GAN-PW, IRecGAN
- Generator can learn the item distribution, and generate fake items
- Discriminator can distinguish real and fake items, and predict user’s feedback

RL-based Recommender Training



- Metric: average reward of a session
- Baselines: Historical Logs, IRecGAN
- UserSim converges to the similar avg_reward with the one upon historical data
- UserSim performs much more stably than the one trained based upon IRecGAN

On-policy RL algorithms such as SARSA cannot be directly trained on historical data

Conclusion

- We propose a novel user simulator based on Generative Adversarial Network
  - Generating real-time feedback like real users
  - Pre-training and evaluating new recommendation algorithms before launching them online

zhaoxi35@msu.edu