# Deep Reinforcement Learning for Page-wise Recommendations

Xiangyu Zhao[1], Long Xia[2], Liang Zhang[2], Zhuoye Ding[2],
Dawei Yin[2], Jiliang Tang[1]

1: Data Science and Engineering Lab, Michigan State University
2: Data Science Lab, JD.com

# Recommender Systems

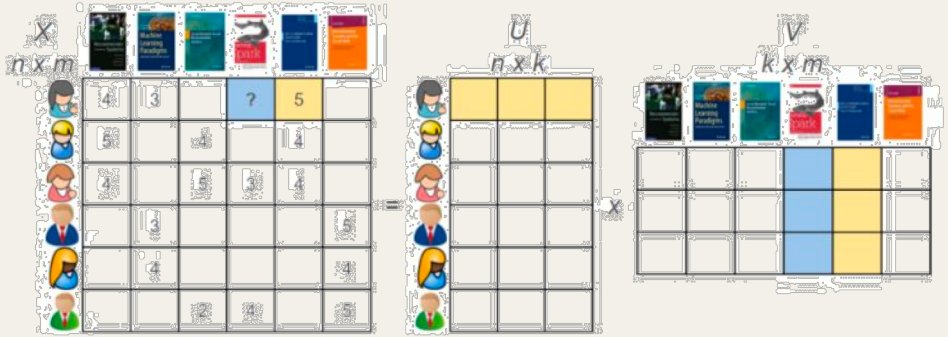
- **Goal: Suggest items that best fit users' preferences**
- **User-System Interactions**

# Challenges

- How to capture user's dynamic preference and update recommending strategy
- How to generate a page of complementary items and display them in a 2-D page

# Existing Recommender Systems

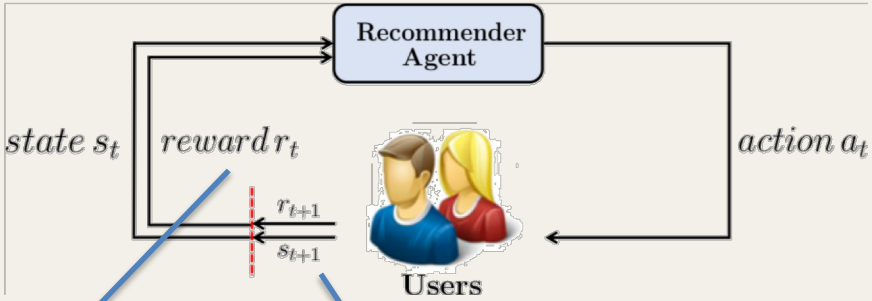
- Recommendation procedure as static process
- Making recommendations following fixed greedy strategy




- Disadvantages
  - Users' dynamic preferences
  - Real-time feedback

# Why Reinforcement Learning?

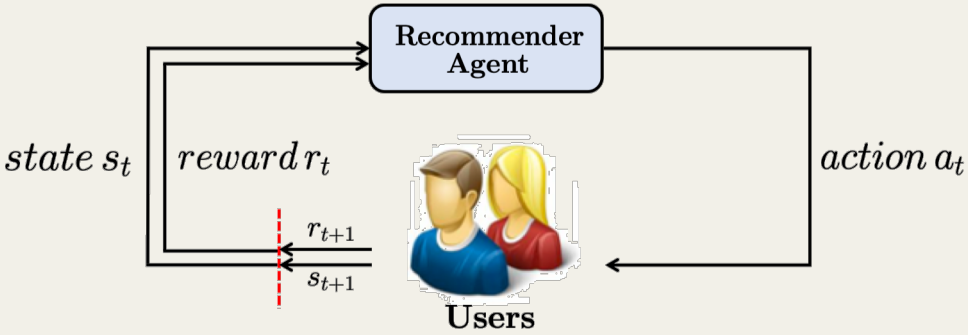
- **Recommendation Procedure**
  - **User-Agent (RA) Sequential Interactions**




  - **RL: Automatically learn the optimal recommendation strategies**

# Why Reinforcement Learning?

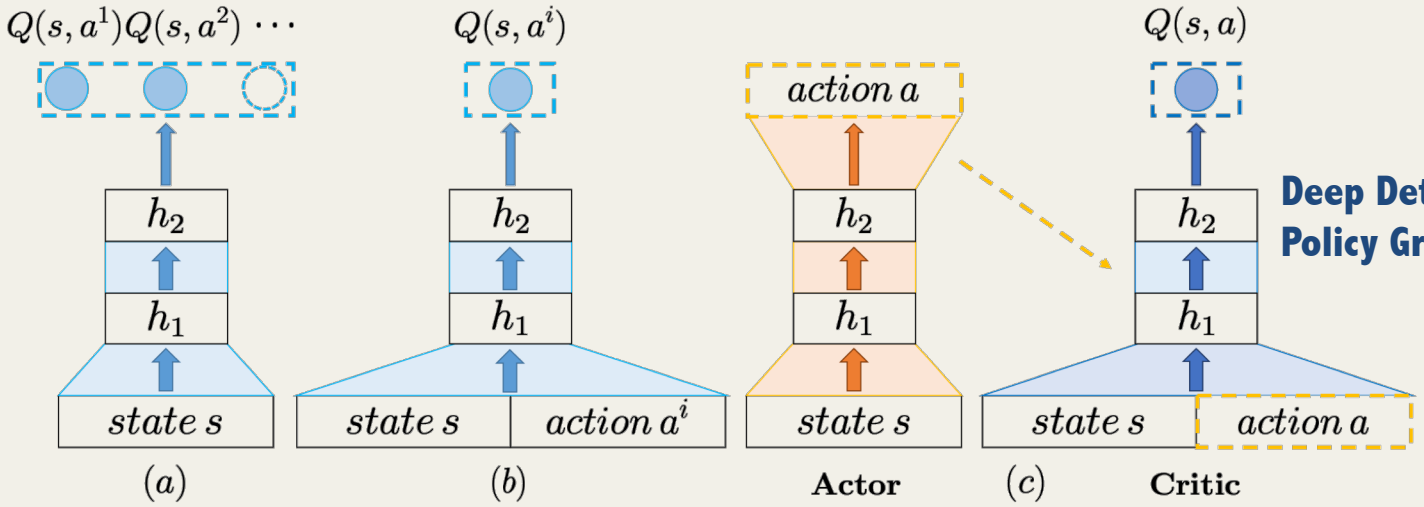
- **Continuously updating the recommendation strategies during the interactions**




- **The optimal strategy is designed to maximize the long-term reward from users**

# RL Architecture Selection

- The large and dynamic action space
- The computational cost to select an optimal action

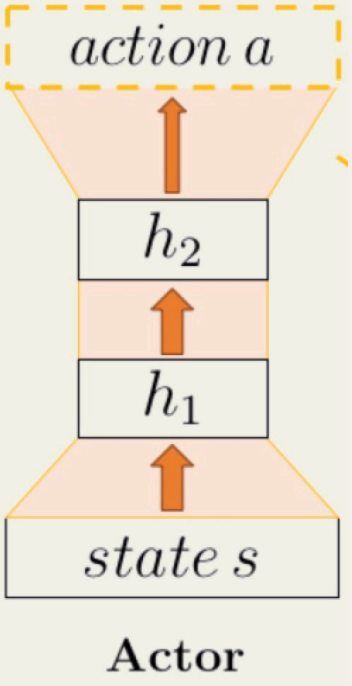



$$Q^*(s,a) = \mathbb{E}_{s'}\left[r + \gamma \max_{a'} Q^*(s',a')|s,a\right] \qquad Q(s,a) = \mathbb{E}_{s'}\left[r + \gamma Q(s',a')|s,a\right]$$

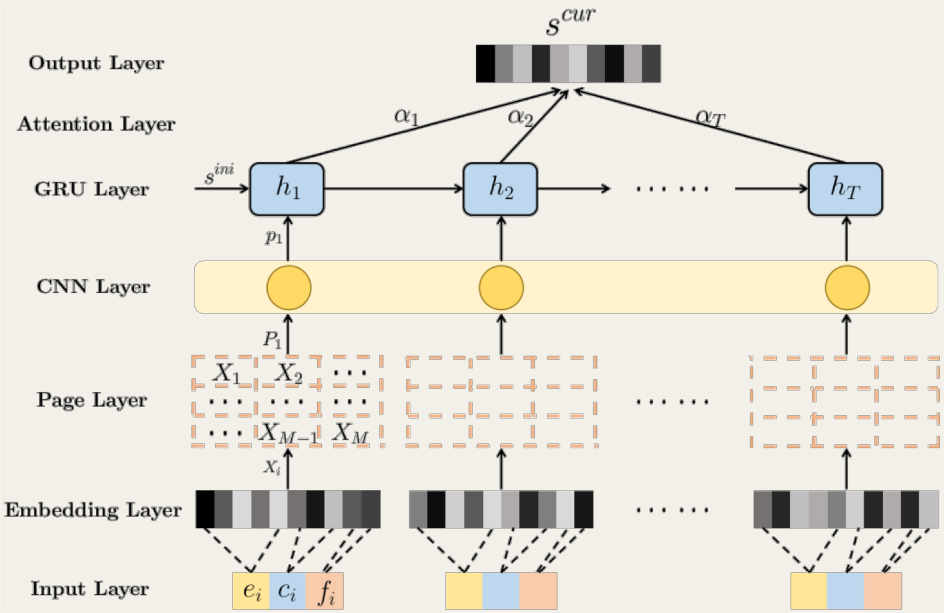
# Actor Design

- **Goal: Generating a page of recommendations according to user's preference**

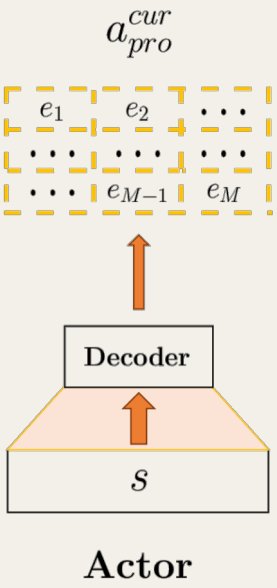
- **Challenges**
  - **Real-time feedback**
  - **A set of complementary items**
  - **Item display on a 2-D page**



# Actor Architecture



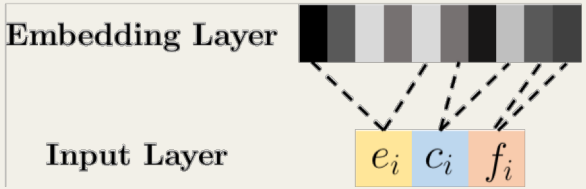

Encoder




Actor

Decoder

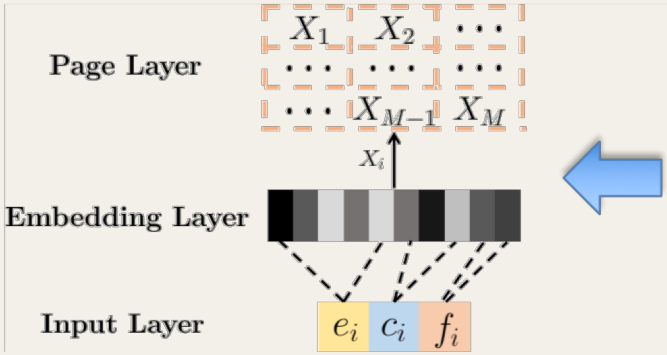
# Embedding and Page Layers

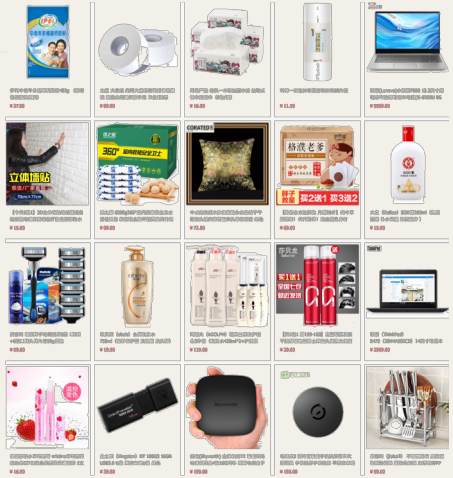



$e_i$: item's embedding

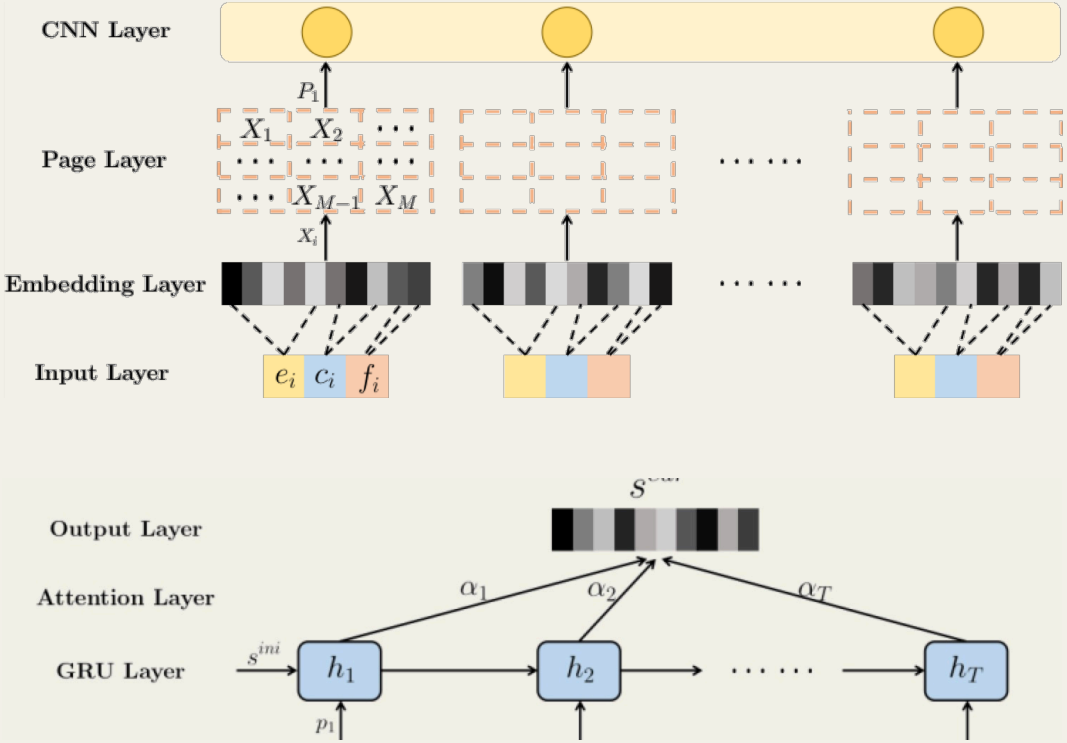
$c_i$: item's category

$f_i$: user's feedback

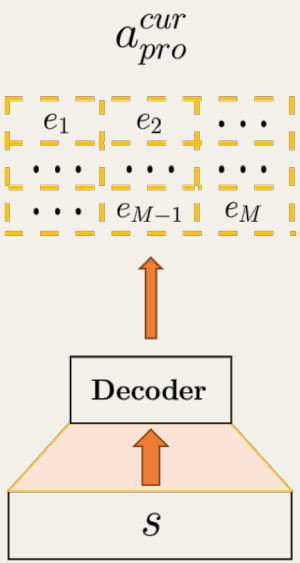


# CNN and RNN Layers



# Decoder

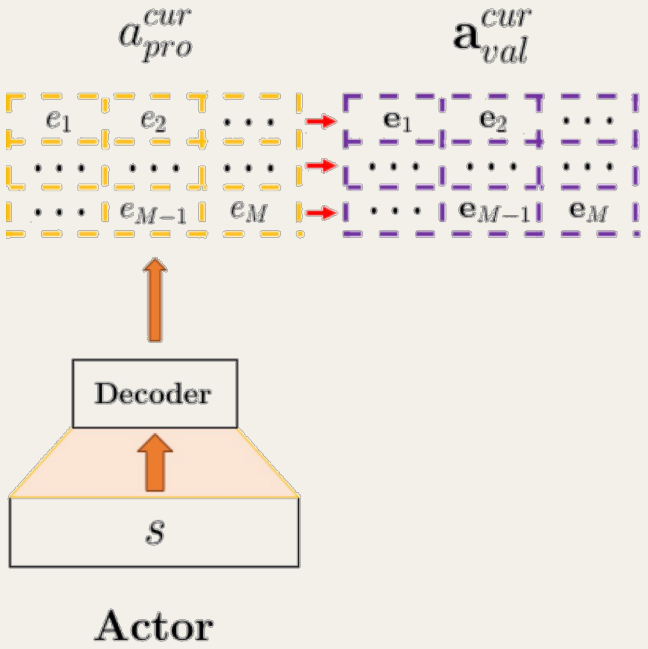



Deconvolution

Neural Network

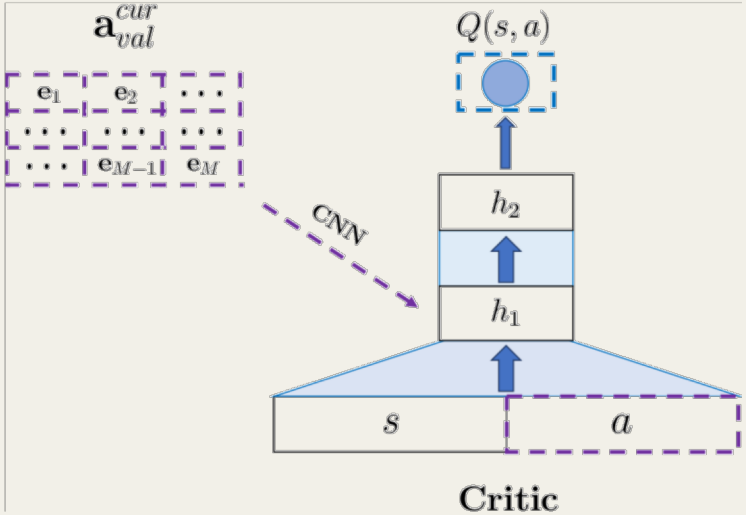
(DeCNN)

**proto-action → valid-action**



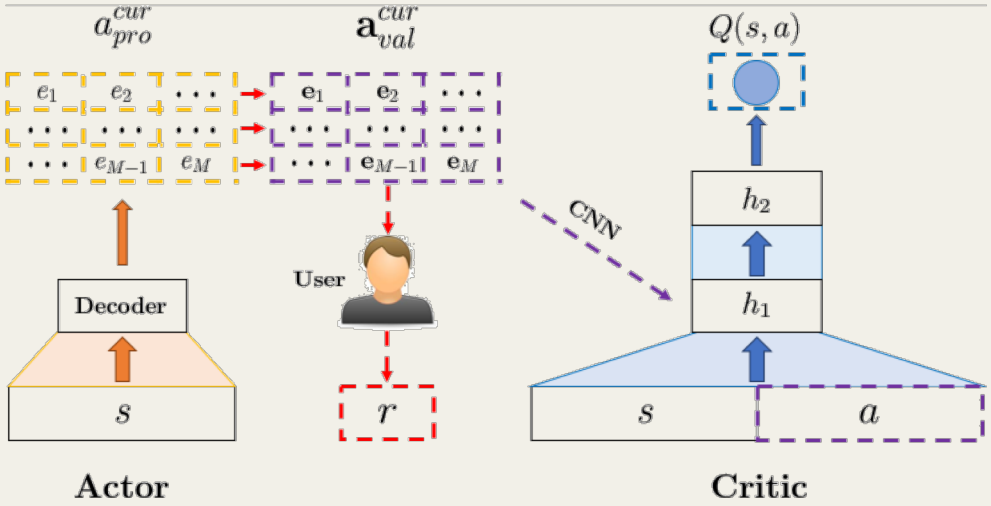
# Critic Architecture

- **Learning action-value function** $Q(s, a)$



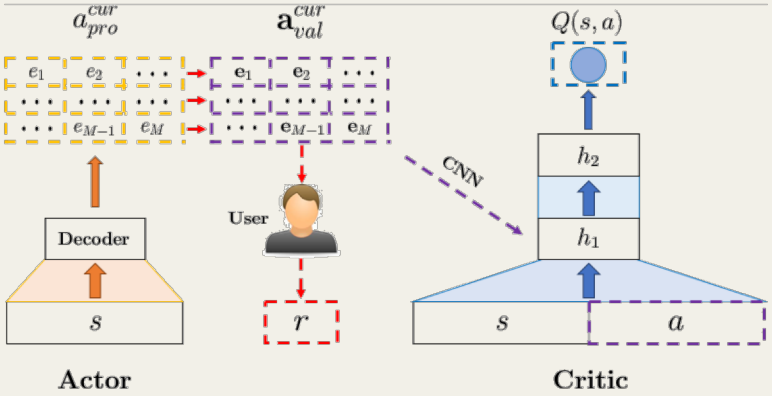
# Online Training Procedure

- **DDPG is utilized to train the framework**




[1] Deep reinforcement learning in large discrete action spaces. arXiv preprint arXiv:1512.07679.

# Offline Training Procedure




- The new recommendation page $\mathbf{a}_{val}^{cur}$ and user's corresponding feedback (reward) $r$ are given in the data

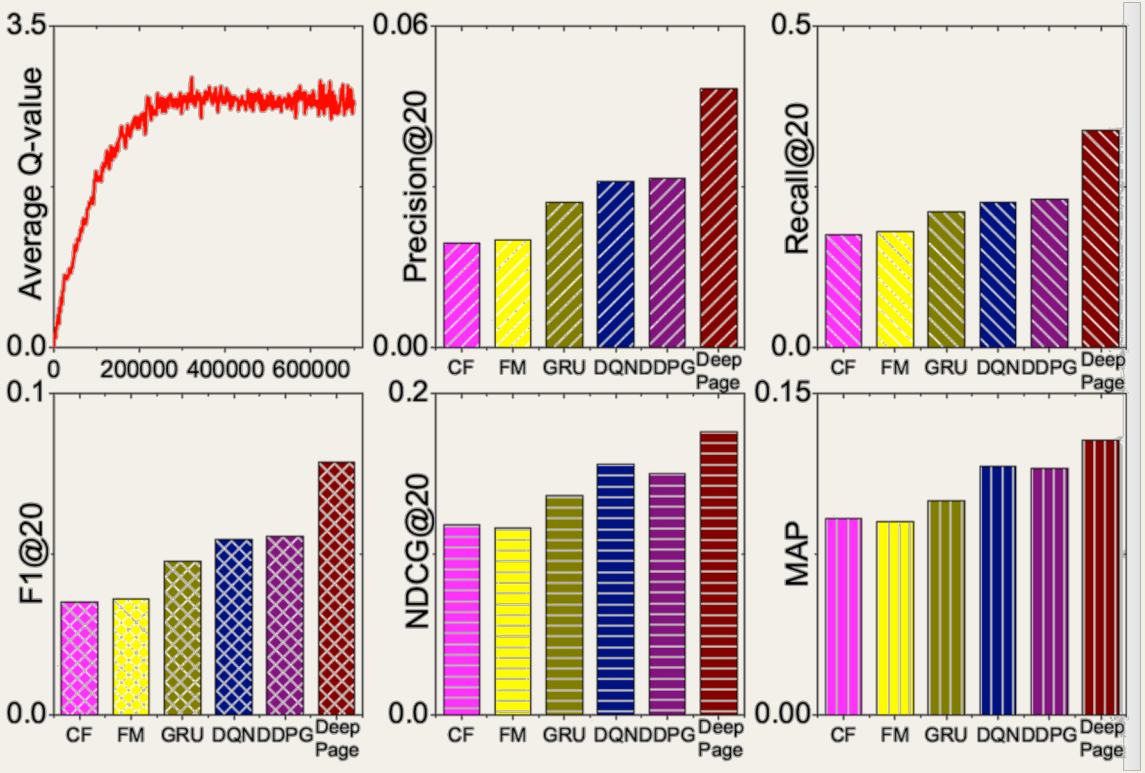
$$\min_{\theta^{\pi}} \sum_{b=1}^{B} \left( \| a_{pro}^{cur} - \mathbf{a}_{val}^{cur} \|_F^2 \right)$$

# Experiment Settings

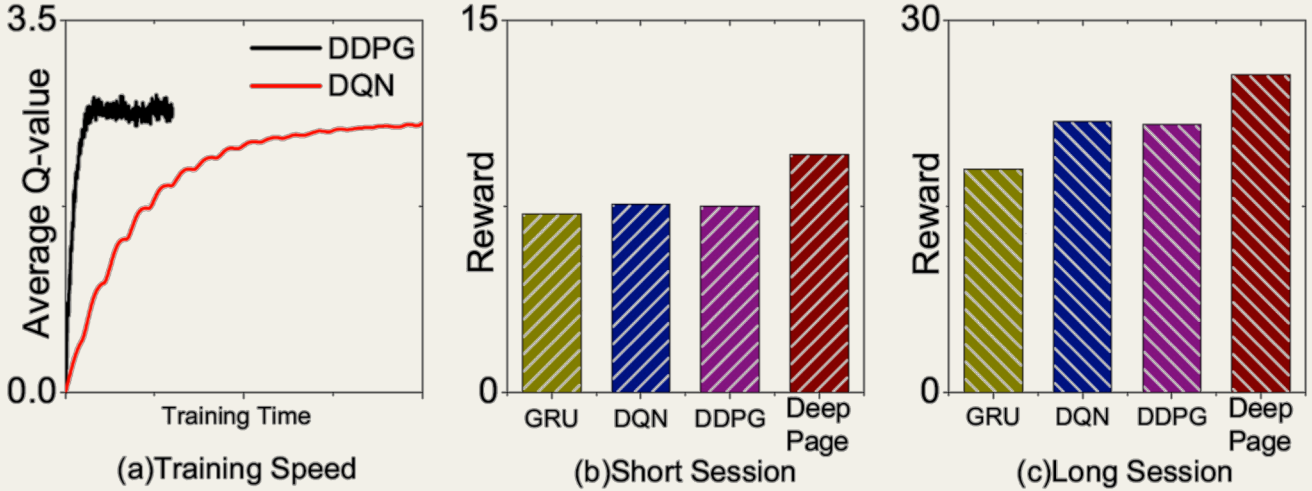
- **Datasets from JD.com**
  - **Online: simulated online environment**
  - **Offline**
    - **1,000,000 recommendation sessions in temporal order**
      - **first 70% sessions as the training/validation set**
      - **later 30% sessions as the test set**
  - **Each time the RA recommends a page of 10 (= 5 x 2) items to users**
  - **Metrics: MAP and NDCG for offline test, Accumulated Reward for online test**

# Performance Comparison for Offline Test

- **Baselines**
  - CF
  - FM
  - GRU
  - DQN
  - DDPG



# Performance Comparison for Online Test



# Effectiveness of Components

- DeepPage-1: no embedding-layers
- DeepPage-2: no one-hot vectors of category and feedback
- DeepPage-3: no GRU to generate initial state
- DeepPage-4: no CNNs
- DeepPage-5: no attention layers
- DeepPage-6: no GRU to generate local state
- DeepPage-7: no DeCNN

| | Precision @20 | Recall @20 | F1-score @20 | NDCG @20 | MAP |
|---|---|---|---|---|---|
| DeepPage-1 | 0.0479 | 0.3351 | 0.0779 | 0.1753 | 0.1276 |
| DeepPage-2 | 0.0475 | 0.3308 | 0.0772 | 0.1737 | 0.1265 |
| DeepPage-3 | 0.0351 | 0.2627 | 0.0578 | 0.1393 | 0.1071 |
| DeepPage-4 | 0.0452 | 0.3136 | 0.0729 | 0.1679 | 0.1216 |
| DeepPage-5 | 0.0476 | 0.3342 | 0.0775 | 0.1716 | 0.1243 |
| DeepPage-6 | 0.0318 | 0.2433 | 0.0528 | 0.1316 | 0.1039 |
| DeepPage-7 | 0.0459 | 0.3179 | 0.0736 | 0.1698 | 0.1233 |
| **DeepPage** | **0.0491** | **0.3576** | **0.0805** | **0.1872** | **0.1378** |

# Future Work

- Handling multiple tasks collectively in one RL framework
  - Search
  - Bidding/Auction
  - Advertisement
  - Recommendation
- Reducing the temporal complexity of mapping from proto-action to valid-action





# Thanks

http://www.cse.msu.edu/~zhaoxi35/
zhaoxi35@msu.edu