Deep Reinforcement Learning for Page-wise Recommendations

Xiangyu Zhao¹, Long Xia², Liang Zhang², Zhuoye Ding², Dawei Yin², Jiliang Tang¹

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

\[ Q(s, a^1)Q(s, a^2) \cdots \]

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

Deep Deterministic Policy Gradient (DDPG)
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

Decoder
Embedding and Page Layers

- $e_i$: item’s embedding
- $c_i$: item’s category
- $f_i$: user’s feedback
CNN and RNN Layers

- **Input Layer**: $e_i$, $c_i$, $f_i$
- **Embedding Layer**
- **Page Layer**: $X_1, X_2, \ldots, X_{M-1}, X_M$
- **CNN Layer**
- **Output Layer**: $s^{\text{out}}$
- **Attention Layer**: $\alpha_1, \alpha_2$
- **GRU Layer**: $h_1, h_2, \ldots, h_T$
Decoder

Deconvolution Neural Network (DeCNN)

proto-action → valid-action
Critic Architecture

- **Learning action-value function** $Q(s, a)$
DDPG is utilized to train the framework.

The new recommendation page \( a_{cur}^{val} \) and user’s corresponding feedback (reward) \( r \) are given in the data.

\[
\min_{\theta, \pi} \sum_{b=1}^{B} (\| a_{pro}^{cur} - a_{val}^{cur} \|^2_F)
\]
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

(a) Training Speed

(b) Short Session

(c) Long Session
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

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