Generative Adversarial User Model for Reinforcement Learning Based Recommendation System

Xinshi Chen¹, Shuang Li¹, Hui Li², Shaohua Jiang², Yuan Qi², Le Song¹,²

¹Georgia Tech, ²Ant Financial
ICML 2019
A user’s interest evolves over time based on what she observes.

Recommender’s action can significantly influence such evolution.

A RL based recommender can consider user’s long term interest.
Challenges

1. User is the **environment** → Training of **RL** policy requires lots of interactions with users

   e.g. (1) For *AlphaGo Zero*, 4.9 million games of self-play were generated for training.
   (2) RL for *Atari* game takes more than **50 hours** on GPU for training.

2. The **reward** function (a user’s interest) is unknown
Our solution

We propose

• A **Generative Adversarial User Model**
  - to model user’s *action*
  - to recover user’s *reward*

• Use GAN User Model as a *simulator* to pre-train the **RL** policy offline
Generative Adversarial User Model

2 components:

User’s reward $r(s^t, a^t)$
- $a^t$ is clicked item.
- $s^t$ is user’s experience (state).

User’s behavior $\phi(s^t, A^t)$
- $A^t$ contains items displayed by the system.
- act $a^t \sim \phi$ to maximize her expected reward.
- $\phi^*(s^t, A^t) = \arg \max \mathbb{E}_\phi [r(s^t, a^t)] - R(\phi)/\eta$
Generative Adversarial Training

In analogy to GAN:

- \( \phi \) (behavior) acts as a generator
- \( r \) (reward) acts as a discriminator

Jointly learned via a \textbf{mini-max formulation}:

\[
\min_r \max_\phi \mathbb{E}_\phi \left[ \sum_{t=1}^{T} r(s_t^t, a_t^t) \right] - R(\phi)/\eta - \sum_{t=1}^{T} r(s_{true}^t, a_{true}^t)
\]
2 architectures for aggregating historical information (i.e. state $s^t$)

(1) LSTM

(2) Position Weight
Set Recommendation RL policy

all available $K$ items

set recommendation

display $k$ items

$a_1^*, a_2^*, \ldots, a_k^* = \arg \max_{a_1, \ldots, a_k} Q(s_t, a_1, a_2, \ldots, a_k)$

combinatorial action space $\binom{K}{k}$  \rightarrow  Intractable computation!
We design a cascading Q network to compute the optimal action with linear complexity:

\[ a_1^*, a_2^*, \ldots, a_k^* = \arg \max_{a_1, \ldots, a_k} Q(s^t, a_1, a_2, \ldots, a_k) \]

\[ a_1^* = \arg \max_{a_1} Q^{1*}(s^t, a_1) \]

\[ a_2^* = \arg \max_{a_2} Q^{2*}(s^t, a_1^*, a_2) \]

\[ \ldots \]

\[ a_k^* = \arg \max_{a_k} Q^{k*}(s^t, a_1^*, a_2^*, \ldots, a_k) \]
Set Recommendation RL policy: Cascading DQN

\[ \begin{align*}
& \text{Argmax } \bar{a}_1 \\
& \text{Argmax } \bar{a}_2 \\
& \text{Argmax } \ldots \\
& \text{Argmax } \bar{s} \bar{a}_1 \bar{a}_2 \ldots \\
& \text{Argmax } Q_1(s, \bar{a}_1; \theta_1) \\
& \text{Argmax } Q_2(s, \bar{a}_1, \bar{a}_2; \theta_2) \\
& \text{Argmax } Q_k(s, \bar{a}_1; \theta_1) \\
\end{align*} \]
### Predictive Performance of User Model

<table>
<thead>
<tr>
<th>Model</th>
<th>MovieLens prec@1 (%)</th>
<th>MovieLens prec@2 (%)</th>
<th>LastFM prec@1 (%)</th>
<th>LastFM prec@2 (%)</th>
<th>Ant Financial prec@1 (%)</th>
<th>Ant Financial prec@2 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>IKNN</td>
<td>38.8 ±1.9</td>
<td>40.3 ±1.9</td>
<td>20.4 ±0.6</td>
<td>32.5 ±1.4</td>
<td>20.6 ±0.2</td>
<td>32.1 ±0.2</td>
</tr>
<tr>
<td>S-RNN</td>
<td>39.3 ±2.7</td>
<td>42.9 ±3.6</td>
<td>9.4 ±1.6</td>
<td>17.4 ±0.9</td>
<td>32.2 ±0.9</td>
<td>40.3 ±0.6</td>
</tr>
<tr>
<td>SCkNNC</td>
<td>49.4 ±1.9</td>
<td>51.8 ±2.3</td>
<td>21.4 ±0.5</td>
<td>26.1 ±1.0</td>
<td>34.6 ±0.7</td>
<td>43.2 ±0.8</td>
</tr>
<tr>
<td>XGBOOST</td>
<td>66.7 ±1.1</td>
<td>76.0 ±0.9</td>
<td>10.2 ±2.6</td>
<td>19.2 ±3.1</td>
<td>41.9 ±0.1</td>
<td>65.4 ±0.2</td>
</tr>
<tr>
<td>DFM</td>
<td>63.3 ±0.4</td>
<td>75.9 ±0.3</td>
<td>10.5 ±0.4</td>
<td>20.4 ±0.1</td>
<td>41.7 ±0.1</td>
<td>64.2 ±0.2</td>
</tr>
<tr>
<td>W&amp;D-LR</td>
<td>61.5 ±0.7</td>
<td>73.8 ±1.2</td>
<td>7.6 ±2.9</td>
<td>16.6 ±3.3</td>
<td>37.5 ±0.2</td>
<td>60.9 ±0.1</td>
</tr>
<tr>
<td>W&amp;D-CCF</td>
<td>65.7 ±0.8</td>
<td>75.2 ±1.1</td>
<td>15.4 ±2.4</td>
<td>25.7 ±2.6</td>
<td>37.7 ±0.1</td>
<td>61.1 ±0.1</td>
</tr>
<tr>
<td>GAN-PW</td>
<td>66.6 ±0.7</td>
<td>75.4 ±1.3</td>
<td><strong>24.1 ±0.8</strong></td>
<td><strong>34.9 ±0.7</strong></td>
<td>41.9 ±0.1</td>
<td>65.8 ±0.1</td>
</tr>
<tr>
<td>GAN-LSTM</td>
<td><strong>67.4 ±0.5</strong></td>
<td><strong>76.3 ±1.2</strong></td>
<td>24.0 ±0.9</td>
<td>34.9 ±0.8</td>
<td><strong>42.1 ±0.2</strong></td>
<td><strong>65.9 ±0.2</strong></td>
</tr>
</tbody>
</table>

### Recommendation Policy Based On User Model

<table>
<thead>
<tr>
<th>Model</th>
<th>reward@k=3</th>
<th>reward@k=5</th>
<th>ctr@k=3</th>
<th>ctr@k=5</th>
</tr>
</thead>
<tbody>
<tr>
<td>W&amp;D-LR</td>
<td>14.46 ±0.42</td>
<td>15.18 ±0.38</td>
<td>0.46 (±0.01)</td>
<td>0.48 (±0.01)</td>
</tr>
<tr>
<td>W&amp;D-CCF</td>
<td>19.93 ±1.09</td>
<td>20.94 ±1.03</td>
<td>0.62 (±0.03)</td>
<td>0.65 (±0.03)</td>
</tr>
<tr>
<td>GAN-Greedy</td>
<td>21.37 ±1.24</td>
<td>22.97 ±1.22</td>
<td>0.67 (±0.04)</td>
<td>0.71 (±0.03)</td>
</tr>
<tr>
<td>GAN-RWD1</td>
<td>22.17 ±1.07</td>
<td>25.15 ±1.04</td>
<td>0.68 (±0.03)</td>
<td><strong>0.78</strong> (±0.03)</td>
</tr>
<tr>
<td>GAN-GDQN</td>
<td>23.60 ±1.06</td>
<td>23.19 ±1.17</td>
<td>0.72 (±0.03)</td>
<td>0.70 (±0.03)</td>
</tr>
<tr>
<td><strong>GAN-CDQN</strong></td>
<td><strong>24.05</strong> (±0.98)</td>
<td><strong>25.36</strong> (±1.10)</td>
<td><strong>0.74</strong> (±0.03)</td>
<td><strong>0.77</strong> (±0.03)</td>
</tr>
<tr>
<td>DQN-Off</td>
<td>20.31 ±0.14</td>
<td>21.82 ±0.08</td>
<td>0.63 (±0.01)</td>
<td>0.67 (±0.01)</td>
</tr>
</tbody>
</table>
Experiments

Cascading-DQN policy *pre-trained* over a **GAN User Model** can quickly achieve a high CTR even when it is applied to a new set of users.

![Graphs showing click rate over number of interactions for different policies](image)

*Figure*: Comparison of the averaged click rate averaged over 1,000 users under different recommendation policies. *X*-axis represents how many times the recommender interacts with online users. *Y*-axis is the click rate. Each point \((x, y)\) means the click rate \(y\) is achieved after \(x\) times of user interactions.
Thanks!

Poster: Pacific Ballroom #252, Tue, 06:30 PM

Contact: xinshi.chen@gatech.edu