Intention-aware Sequential Recommendation with Structured Intent Transition

Haoyang Li
Tsinghua University
Recommender Systems (RS) are Ubiquitous

- A day in our life with Recommender Systems

07:00

07:40

09:00

10:00

12:00

14:00

16:00

17:00

20:00
Problem of Today’s RS: User Intentions

- Can RS understand the **user intentions** behind the **behaviors**?
- Behaviors are highly driven by user intentions in the real world.

### Example

#### Multipurpose - Our multipurpose bat
- **Play baseball**
- **Gift**
- **Self-defense**
- **Attack**
A Possible Solution: Learn from Sequence

- Identifying user intentions exactly from one item is difficult, but may be possible from a sequence of items (behaviors).

self-drive tour
Can RS model the transition of user intentions explicitly?

- In the real world, user intentions could be dynamic rather than static.
- “dynamic” : “intent transition”

Problem of Today’s RS: User Intentions
Sequence Modeling

- **Sequential Recommendation**
  - Order matters in real-world situations.

- **Sequence Modeling**
  - **Markov Chain**: fails on long sequences, data sparsity problem
  - **Recurrent Neural Network**: fails on longer sequences, high cost
  - **Transformer**
Modeling User Intentions is Challenging

- The existing methods capture behavior patterns. They fail to identify user intentions and model intent transition explicitly.

- The properties of user intentions in Recommendation:
  - Unobserved
  - Multiple
  - Correlated
  - Dynamic
  - auxiliary information
  - identify
  - cognitive relations
  - intent transition
Problem

- Intention-aware Sequential Recommendation
  - Goal: for each user, given the sequence of interacted items $v_1, ..., v_t$ with available description information, predict the $v_{t+1}$ at time index $t + 1$.
  - Description information: title, categories, reviews...

- We extract keywords from the description information and refer to these extracted keywords as concepts.
  - User intentions could be reflected in these concepts.
  - E.g.,
    - “Yes, it’s perfect for self-defense.” → “perfect” “self-defense”
    - “This might be great for my son playing baseball.” → “great” “son” “baseball”
Model Framework

1. Transformer-based Encoder

2. Intent Extraction

3. Structured Intent Transition

4. Intent Decoder
Transformer-based Encoder

- **Embedding Submodule**
  - Embedding \( h_i \) = item \( v_i \) + position \( p_i \) + concepts \( c_j \)
    - Item Property  |  Order  |  Potential Intents

- Input hidden representation: \( H^0 = [h_1^0, h_2^0, ..., h_T^0] \quad h_i = v_i + p_i + \sum_{c_{i,j}=1} c_j \)

- **Self-attention Submodule**
  - Capture the dependencies among items within a behavior sequence

\[
S^l = \text{SA}(H^l) = \text{Attention}(H^lW_Q^l, H^lW_K^l, H^lW_V^l) \\
H^{l+1} = \text{FFN}(S^l) = \text{ReLU}(S^lW_1^l + b_1^l)W_2^l + b_2^l
\]

- Goals: Learn users’ behavior patterns and filter some noises
Intent Extraction

In this step, we explicitly extract explainable user intents from the encoded sequence hidden representations $X = H^L$.

- Goal: Infer a multi-hot intention vector $m_t = [m_{t,1}, ..., m_{t,K}]$
  - $m_{t,k} = 1 \Leftrightarrow k^{th}$ concept belongs to the user intentions at time $t$

A straightforward method: treating $m_t$ as a parameter to be optimized? (×)
- over-parameterization; cause efficiency burdens

We adopt a prototype-based method:
- define $K$ intention prototypes, calculate similarity $s_{t,k} = \frac{x_t \cdot c_k}{\|x_t\|_2 \|c_k\|_2}$
- draw $m_t$ from $m_t \sim \text{Categorical}($Softmax$(s_{t,1}, s_{t,2}, ..., s_{t,K}))$
Structured Intent Transition

- In this step, we model intent transitions with GNN.
  - Learn a personalized intent feature matrix \( Z_t = [z_{t,1}, \ldots, z_{t,K}] \in \mathbb{R}^{K \times d'} \)
  - From item space to intention space \( z_{t,k} = m_{t,k} \text{MLP}_k(x_t) \)
  - Model the intent transition on the concept graph \( A \) (ConceptNet)
    - Pre-defined concepts and their relations can be treated as knowledge; underlying cognitive activity is stored in the connections among concepts.
  - We adopt the message-passing framework \( Z_{t+1} = \mathcal{F}(Z_t, A) \)
Intent Decoder

- After obtaining the future intent features $Z_{t+1}$ and intent vector $m_{t+1}$, the intent decoder is defined as:

$$x_{t+1} = \sum_{k=1}^{K} m_{t+1,k} \text{MLP}'_k(z_{t+1,k})$$

- Recommendation probability of item $v_{t+1}$:

$$p(v_{t+1}|v_1, v_2, \ldots, v_t) = \text{Softmax}(x_{t+1}V^T)$$

- Objective Function

$$\mathcal{L}_u = \frac{1}{|\mathcal{S}(u)|} \sum_{v_{t+1} \in \mathcal{S}(u)} -\log p(v_{t+1}|v_1, v_2, \ldots, v_t)$$
Experiments

- We aim to answer the following three questions:
  - Q1: How does our method perform compared with other state-of-the-art sequential recommendation methods?
  - Q2: Can our method identify explainable user intents and model the structured intent transition accurately?
  - Q3: Is the intent extraction and structured intent transition module helpful in our method?
Experimental Settings

- **Datasets**
  - **Amazon**: contains a large number of product reviews from Amazon.com
    - We choose the “Beauty” category dataset.
  - **Steam**: a popular online video game platform
  - **Epinions**: a popular online consumer review website Epinions.com
  - **MovieLens**: a dataset about movie rating, including ML-1m an ML-20m
Experimental Settings

- Datasets preprocessing procedures
  - convert all reviews/ratings to implicit feedback of 1
  - remove users and items if they have fewer than 5 records
  - build the interaction sequence sorted by the timestamps for each user
  - obtain concepts of items from the available meta-data
    - e.g., items’ descriptions, reviews, etc.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Users</th>
<th>#Items</th>
<th>#Interactions</th>
<th>Avg.length</th>
<th>Density</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beauty</td>
<td>40,226</td>
<td>54,542</td>
<td>0.35m</td>
<td>8.8</td>
<td>0.02%</td>
</tr>
<tr>
<td>Steam</td>
<td>281,428</td>
<td>13,044</td>
<td>3.5m</td>
<td>12.4</td>
<td>0.10%</td>
</tr>
<tr>
<td>Epinions</td>
<td>5,015</td>
<td>8,335</td>
<td>26.9k</td>
<td>5.37</td>
<td>0.06%</td>
</tr>
<tr>
<td>ML-1m</td>
<td>6,040</td>
<td>3,416</td>
<td>1.0m</td>
<td>163.5</td>
<td>4.79%</td>
</tr>
<tr>
<td>ML-20m</td>
<td>138,493</td>
<td>26,744</td>
<td>20m</td>
<td>144.4</td>
<td>0.54%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Concepts</th>
<th>#Edges</th>
<th>Avg.concepts/item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beauty</td>
<td>592</td>
<td>2,791</td>
<td>4.45</td>
</tr>
<tr>
<td>Steam</td>
<td>229</td>
<td>472</td>
<td>4.49</td>
</tr>
<tr>
<td>Epinions</td>
<td>114</td>
<td>467</td>
<td>5.50</td>
</tr>
<tr>
<td>ML-1m</td>
<td>96</td>
<td>327</td>
<td>1.94</td>
</tr>
<tr>
<td>ML-20m</td>
<td>316</td>
<td>842</td>
<td>4.21</td>
</tr>
</tbody>
</table>
Experimental Settings

- **Evaluation metrics**

- **Hit Rate (HR)**

  \[
  HR@k = \frac{1}{|U|} \sum_{u \in U} \delta(|T_u \cap R_{u,k}| > 0)
  \]

- **Normalized Discounted Cumulative Gain (NDCG)**

  \[
  NDCG@k = \frac{1}{Z} DCG@k = \frac{1}{Z} \frac{1}{|U|} \sum_{u \in U} \sum_{i=1}^{k} \frac{\delta(r_{u,i} \in T_u)}{\log_2(i + 1)}
  \]

- **Mean Reciprocal Rank (MRR)**

  \[
  MRR = \frac{1}{|U|} \sum_{u \in U} \frac{1}{\text{rank}_u}
  \]
Baselines

- Non-sequential methods
  - PopRec, BPR-MF, NCF

- Sequential methods
  - Markov chain based methods: FPMC, Caser
  - RNN based methods: GRU4Rec, GRU4Rec+
  - Transformer based methods: SASRec, BERT4Rec
## Experimental Results

### Recommendation Accuracy

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Metric</th>
<th>PopRec</th>
<th>BPR-MF</th>
<th>NCF</th>
<th>FPMC</th>
<th>GRU4Rec</th>
<th>GRU4Rec+</th>
<th>DGC</th>
<th>Caser</th>
<th>SASRec</th>
<th>BERTRec</th>
<th>ISRec</th>
<th>Improv.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beauty</td>
<td>HR@1</td>
<td>0.0077</td>
<td>0.0415</td>
<td>0.0407</td>
<td>0.0435</td>
<td>0.0402</td>
<td>0.0551</td>
<td>0.0626</td>
<td>0.0475</td>
<td>0.0906</td>
<td>0.0953</td>
<td>0.1233</td>
<td>29.38%</td>
</tr>
<tr>
<td></td>
<td>HR@5</td>
<td>0.0392</td>
<td>0.1209</td>
<td>0.1305</td>
<td>0.1387</td>
<td>0.1315</td>
<td>0.1781</td>
<td>0.1835</td>
<td>0.1625</td>
<td>0.1934</td>
<td>0.2207</td>
<td>0.2734</td>
<td>23.88%</td>
</tr>
<tr>
<td></td>
<td>HR@10</td>
<td>0.0762</td>
<td>0.1992</td>
<td>0.2142</td>
<td>0.2401</td>
<td>0.2343</td>
<td>0.2654</td>
<td>0.2778</td>
<td>0.2590</td>
<td>0.2653</td>
<td>0.3025</td>
<td>0.3594</td>
<td>18.81%</td>
</tr>
<tr>
<td></td>
<td>NDCG@5</td>
<td>0.0230</td>
<td>0.0184</td>
<td>0.0855</td>
<td>0.0902</td>
<td>0.0902</td>
<td>0.1172</td>
<td>0.1214</td>
<td>0.1091</td>
<td>0.1346</td>
<td>0.1599</td>
<td>0.2020</td>
<td>26.33%</td>
</tr>
<tr>
<td></td>
<td>NDCG@10</td>
<td>0.0349</td>
<td>0.1064</td>
<td>0.1124</td>
<td>0.1211</td>
<td>0.1074</td>
<td>0.1453</td>
<td>0.1543</td>
<td>0.1360</td>
<td>0.1633</td>
<td>0.1682</td>
<td>0.2296</td>
<td>23.31%</td>
</tr>
<tr>
<td></td>
<td>MRR</td>
<td>0.0437</td>
<td>0.1000</td>
<td>0.1043</td>
<td>0.1056</td>
<td>0.1023</td>
<td>0.1299</td>
<td>0.1381</td>
<td>0.1205</td>
<td>0.1536</td>
<td>0.1701</td>
<td>0.2081</td>
<td>22.34%</td>
</tr>
</tbody>
</table>

### Steam

- **Recommendation Accuracy**
- **Self-attention can provide large performance gains.**
- **Our method outperforms all the baselines.**
- **The sparser the dataset, the larger the improvement of our method.**

### Epinions

### ML-1m

### ML-20m
Showcases of Intent Extraction and Structured Intent Transition
Experimental Results

- Ablation studies and hyperparameters sensitivities

**TABLE 5: Performance comparison of ISRec and variants.**

<table>
<thead>
<tr>
<th></th>
<th>Beauty HR@10</th>
<th>Beauty NDCG@10</th>
<th>ML-1m HR@10</th>
<th>ML-1m NDCG@10</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISRec</td>
<td>0.3594</td>
<td>0.2296</td>
<td>0.7363</td>
<td>0.5189</td>
</tr>
<tr>
<td>w/o GNN</td>
<td>0.3311</td>
<td>0.2095</td>
<td>0.7222</td>
<td>0.4978</td>
</tr>
<tr>
<td>w/o GNN&amp;Intent</td>
<td>0.3092</td>
<td>0.1965</td>
<td>0.7058</td>
<td>0.4731</td>
</tr>
<tr>
<td>BERT4Rec + concept</td>
<td>0.3037</td>
<td>0.1886</td>
<td>0.6987</td>
<td>0.4824</td>
</tr>
<tr>
<td>SASRec + concept</td>
<td>0.3061</td>
<td>0.1845</td>
<td>0.6972</td>
<td>0.4643</td>
</tr>
</tbody>
</table>
Conclusions

- We study the intent-aware sequential recommendation with structured intent transition.
- Why can’t the current Sequential Recommender Systems make us satisfied enough?

知其然，而不知其所以然

User Behaviors  User Intentions

Modeling user intentions with GNN:
Try to promote human-like Recommender Systems.
References

Intention-aware Sequential Recommendation with Structured Intent Transition

Haoyang Li; Xin Wang; Ziwei Zhang; Jianxin Ma; Peng Cui; Wenwu Zhu. Intention-aware Sequential Recommendation with Structured Intent Transition. IEEE TKDE, 2021.
Thanks!

Haoyang Li, Tsinghua University
lihy18@mails.tsinghua.edu.cn

ModelFramework

Experim entalSettings

- Datasets
  - Amazon: contains a large number of product reviews from Amazon.com
  - We choose the "Beauty" category dataset.
  - Steam: a popular online video game platform
  - Epinions: a popular online consumer review website Epinions.com
  - MovieLens: a dataset about movie rating, including ML-1m an ML-20m

Conclusions

- Why can't the current Sequential Recommender Systems make us satisfied enough?

     [Diagram of user behaviors and intentions]

 Modeling user intentions with GNN:
 Try to promote human-like Recommender Systems.