Temporal Learning and Sequence Modeling for a Job Recommender System

Kuan Liu, Xing Shi, Anoop Kumar, Linhong Zhu, and Prem Natarajan

Team: Milk Tea  Presenter: Huaigu Cao
Personalized Job Post Recommendation

Task:

- To recommend job posts to users on Xing,
- based on 1) interaction history and 2) user/item features.

Challenges:

- Large volume
  - 1.5M users, 1.3M items, 8.8M interactions, 200M impression
- Rich/Noisy user/item features available.
  - eps. categorical features. e.g. >100K text tokens
- Temporal dynamics/sequence form in interaction history.
Challenges (cont.)

- **Temporal Dynamics**: *Time as a factor to influence a user's future behavior.*
  - Observation: users tend to re-interact with items that they did in the past.
    - e.g. on average 2 of 7 items in a user’s Week 45 appeared in his past interaction list.
  - Observation: users are more influenced by what they interacted recently than long time ago.

*How to explicitly model temporal influence?*

- **Sequence Property**.
  - User-Item interactions are NOT i.i.d. Instead, a user interacts with a *sequence of items*.
  - Conjecture: Item sequence may contain additional useful information that helps improvement recommender systems. (e.g. temporal relation, item-item similarities.)

*Does sequence really help? If so, how to model?*
Approach Overview

- Temporal-based Ranking on History
- Temporal Matrix Factorization
- Encoder-Decoder Sequence Model
- Ensemble
Approach Overview

● **Temporal Learning**
  ○ A. Temporal based Ranking
  ○ B. Temporal MF

  *Wins over non-temporal counterpart significantly.*

● **Sequence Modeling**
  ○ C. LSTM based Encoder-Decoder model.

  *Beats the best MF model.*
A. Temporal Ranking on Historical Items

Motivation:

- Users have a strong tendency to re-interact with items that they already did in the past.
- More recent interactions influence a user’s future behavior more.

*Historical items are important! Recency of interaction matters!*

Approach:

- A (time reweighted) linear ranking model.
- Minimize a loss incurred on carefully constructed triplet constraints.
A. Temporal Ranking on Historical Items (cont.)

Linear Ranking Model

\[ S(u, i, t) = w M_{u,i,t}^T \]

\[ M_{u,i,t} \in \mathbb{N}^{K \times T} \]

\( w(k, \tau) \) indicates the relative contribution of k-type interaction at time \( \tau \).

Model solving based on triplet constraints

\[ \mathcal{T} = \{ u \text{ prefers to re-interacting with } i_1 \text{ to } i_2 \text{ at time } \tau \}^{N}_{n=1} \]

The distribution between training and test stages as similar as possible!

Construct such constraints when \( u \) interacted with \( i_1, i_2 \) before \( t \), but only interacted with \( i_1 \) at \( t \).
B. Temporal Matrix Factorization

- **Matrix Factorization**
  - To recommend new items

- **Hybrid Matrix Factorization (HMF)**
  - Learn categorical features

- **Temporal HMF (THMF)**
  - Re-weight loss of HMF by time
Hybrid Matrix Factorization (recap)

Users/Items are represented as sums of feature embedding. (b: bias.)

\[ \vec{q}_u = \sum_{j \in f_u} \vec{x}_j^U, \quad \vec{q}_i = \sum_{j \in f_i} \vec{x}_j^I; \quad b_u = \sum_{j \in f_u} b_j^U, \quad b_i = \sum_{j \in f_i} b_j^I \]

User-item score is given by inner product

\[ S(u, i) = \vec{q}_u \cdot \vec{q}_i + b_u + b_i \]

Model is trained by minimizing the loss (we chose WARP) based on score and ground truth \( t \)

\[ L = \sum_{\{u, i\} \in I} \ell(S(u, i), t(u, i)) \]
Temporal Hybrid Matrix Factorization

A non-negative weight associated with time is placed in the loss

\[ L' = \sum_{\{u, i, \tau\} \in I} \ell(S(u, i), t(u, i, \tau)) \times \gamma(\tau) \]

\( \gamma(\tau) \) captures contribution of interactions over time. Zero weights in \( \gamma(\tau) \) reduce training set size as well.

- Value of \( \gamma(\tau) \).
  - in general can be learned jointly with other embedding parameters.
  - in our experiment \textbf{are fixed as learned weights} in Model A. (to speed up training) and give good performance.
C. Sequence Modeling

- **Sequence of items ordered by time:**
  
  - **USER 1:** ITEM 93, ITEM 5, …, ITEM 27 (→ ??, ??, ??)
  - **USER 8:** ITEM 55, ITEM 24, …, ITEM 5 (→ ??, ??, ??)
  - …
  - **USER 65:** ITEM 47, ITEM 7, …, ITEM 62 (→ ??, ??, ??)

- **Tools:**
  
  - Encoder(users)-Decoder(items) framework: next item recommendation is based on both user and previous items.
  - LSTM to model ‘user encoding’ and ‘item transition’.
  - Embedding layer to incorporate feature learning.
Implementation
Important model designs

● **Features**
  ○ Continuous embedding is used to learn categorical features.
  ○ New layer (look-up table and concatenation) is used to connect input and RNN cells.

● **Anonymous users**
  ○ *Item IDs* are treated as categorical features.
  ○ *User IDs are removed* to prevent overfitting.

● **Sampling and data augmentation**
  ○ *No sampling*.
  ○ Original sequence gives better empirical results.
Experiments

Settings:

- 26 to 44 week as training data. 45 as validation.
- Validations are reported.
  - Submitting quota limit
  - Consistent validation/test scores

Evaluation metric:

- Score (all): The challenge score.
- Score(new): The score after removing all user-item pair in the history.
Recommend from history

Scores (in thousands) only based on historical items.

<table>
<thead>
<tr>
<th>Models</th>
<th>Rand</th>
<th>TSort</th>
<th>TRank</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTS</td>
<td>266</td>
<td>284</td>
<td>299</td>
</tr>
<tr>
<td>IMPS</td>
<td>324</td>
<td>375</td>
<td>380</td>
</tr>
<tr>
<td>INTS+IMPS</td>
<td>463</td>
<td>509</td>
<td>524</td>
</tr>
</tbody>
</table>

(The higher, the better.)
Weights associated with time/interaction types.

(a) interactions

(b) impressions
Temporal HMF Improves HMF

Scores (all)

Scores (new)

Latent factor dimensions

Scores

Latent factor dimensions

HMF, no feat.

THMF, no feat.

HMF, w/ feat.

THMF, w/ feat.
THMF Reduces Training Time
Recommend via LSTMs

Performance comparison.

- HMF
- THMF
- LSTM

(The higher, the better.)
Does sequence help?

**Implicit assumption:** sequence or order provides additional information beyond that provided by item frequency alone.

**Experiment:**

- Original sequence.
- Sub-sequence sampling.
Does sequence help?

Implicit assumption: sequence or order provides additional information beyond that provided by item frequency alone.

Experiment:

- Original sequence.
- Sub-sequence sampling.
Conclusion

Our empirical study verifies the effectiveness of

1) utilizing historical information in predicting users’ preferences
2) both temporal and sequence modeling in improving recommendation

Notably, the proposed RNN-based model outperforms the commonly used matrix factorization models.

Future research includes RNN model designs (e.g. to incorporate feature learning in the output layer) and analysis why and when sequence modeling helps recommendation.
Thanks you!
Other slides
Outline

● RecSys Challenge 2016
● Approach Overview
● Temporal Learning
● Sequence modeling
● Experiments
● Conclude
## Recommend via MF

<table>
<thead>
<tr>
<th>Models</th>
<th>Fea</th>
<th>d</th>
<th>score&lt;sub&gt;all&lt;/sub&gt;</th>
<th>score&lt;sub&gt;new&lt;/sub&gt;</th>
<th>T</th>
<th>score&lt;sub&gt;all&lt;/sub&gt;</th>
<th>score&lt;sub&gt;new&lt;/sub&gt;</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
<td>16</td>
<td>235</td>
<td>61</td>
<td>8.8</td>
<td></td>
<td>269</td>
<td>65</td>
<td>2.8</td>
</tr>
<tr>
<td></td>
<td>32</td>
<td>301</td>
<td>71</td>
<td>3.4</td>
<td></td>
<td>320</td>
<td>75</td>
<td>1.5</td>
</tr>
<tr>
<td></td>
<td>48</td>
<td>313</td>
<td>78</td>
<td>7.7</td>
<td></td>
<td>326</td>
<td>84</td>
<td>1.7</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>330</td>
<td>76</td>
<td>3.3</td>
<td></td>
<td>340</td>
<td>86</td>
<td>0.7</td>
</tr>
<tr>
<td>Yes</td>
<td>16</td>
<td>311</td>
<td>124</td>
<td>74</td>
<td></td>
<td>361</td>
<td>146</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td>32</td>
<td>326</td>
<td>125</td>
<td>26</td>
<td></td>
<td>381</td>
<td>148</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>48</td>
<td>354</td>
<td>128</td>
<td>76</td>
<td></td>
<td>378</td>
<td>144</td>
<td>12</td>
</tr>
</tbody>
</table>
## Recommend via LSTMs

<table>
<thead>
<tr>
<th>Fea</th>
<th>No</th>
<th></th>
<th></th>
<th>Yes</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Models</td>
<td>HMF</td>
<td>THMF</td>
<td>LSTM</td>
<td>HMF</td>
<td>THMF</td>
<td>LSTM</td>
</tr>
<tr>
<td>score\textsubscript{all}</td>
<td>313</td>
<td>347</td>
<td>313</td>
<td>312</td>
<td>366</td>
<td>391</td>
</tr>
<tr>
<td>score\textsubscript{new}</td>
<td>78</td>
<td>87</td>
<td>89</td>
<td>104</td>
<td>130</td>
<td>140</td>
</tr>
</tbody>
</table>
Final score before/after ensemble

<table>
<thead>
<tr>
<th>Component</th>
<th>History</th>
<th>MF (ints+imps)</th>
<th>LSTMs</th>
<th>Ensemble</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valid</td>
<td>524</td>
<td>438</td>
<td>391</td>
<td>613</td>
</tr>
<tr>
<td>Test</td>
<td>502</td>
<td>441</td>
<td>384</td>
<td>615</td>
</tr>
</tbody>
</table>