Recommendations Based on Path Predict and LINE

D5: Yu Jia & Youshan Zhuang
May 9th, 2016
Outline

- Background: the need of Recommender systems is surging
- Task Description
- Methodology
  - Information Network & PathSim
  - Graph Embedding & LINE
- Experiments
- Reflections
Background

Good Recommendations

- Attract users
- Bring profits
Our Task

- **Recommend books** to “douban” users
- Douban is a comprehensive entertainment website very popular in China
  - Movie Ratings (0-5 stars)
  - Book Ratings (0-5 stars)
  - Music Ratings (0-5 stars)
  - Users need recommendations
PathSim & Information Network

- Data to heterogeneous information network
- Meta-path: e.g. APVPA
- $s(x, y) = \frac{2 \times |\{p_{x \rightarrow y} : p_{x \rightarrow y} \in \mathcal{P}\}|}{|\{p_{x \rightarrow x} : p_{x \rightarrow x} \in \mathcal{P}\}| + |\{p_{y \rightarrow y} : p_{y \rightarrow y} \in \mathcal{P}\}|}$
- In Our Case: schema; meta-paths on scores
- Many other meta-paths: on music, movies, combinations, etc
Why Graph Embedding?

- Graph Embedding: each node in the graph is given a vector representation
- PathSim may lose information: limited by the length of the meta-paths
- Graph embedding on the other hand
  - Direct relationship between each pair of nodes
  - Abundant information
LINE: Large-scale Information Network Embedding

- First Order Similarity: Nodes with strong ties
- Second Order Similarity: Nodes Share neighbors
- Combine 1st order and 2nd order similarity
- Generate vector representation for each node in the graph

Nodes 6 & 7: high 1st order similarity
Nodes 5 & 6: high 2nd order similarity
Experiment
Path Predict Baseline - redefinition on ratings

What to recommend to User1?

<table>
<thead>
<tr>
<th>Ratings</th>
<th>Book1</th>
<th>Book2</th>
<th>Book3</th>
<th>Book4</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1</td>
<td>5</td>
<td>-</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>User 2</td>
<td>1</td>
<td>5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>User 3</td>
<td>1</td>
<td>1</td>
<td>-</td>
<td>5</td>
</tr>
</tbody>
</table>

$5 \times 1 \times 5 = 25 > 1 \times 1 \times 5 = 5$

Book 2 > Book 4

$5 \times \text{adj55} + \text{adj54} + \text{adj43} + \text{adj32} + \text{adj21}$

only build relations for two ratings that have equal or less than one score difference

No!  Yes!
Procedures

Extract Information from Raw Dataset & Embedding Using LINE

Build up Pearson Correlation Matrix Among Users, Movies, Books and Music Nodes (e.g. user-user, user-book, etc.)

Generate Meta-paths, Treat Each Path as a Feature, Do Matrix Multiplication for each of them

Train these Features with Logistic Regression and Evaluate the Prediction Quality with F1 score for each Meta-Path
## Data Selection

### Raw Data

<table>
<thead>
<tr>
<th>Data Size</th>
<th>user</th>
<th>book</th>
<th>movie</th>
<th>music</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vertices (K)</td>
<td>53.5</td>
<td>147.5</td>
<td>56.5</td>
<td>237.5</td>
</tr>
<tr>
<td>Edges (K)</td>
<td>-</td>
<td>1355</td>
<td>6516</td>
<td>3036</td>
</tr>
</tbody>
</table>

### Train Data

<table>
<thead>
<tr>
<th>Data Size</th>
<th>user</th>
<th>book</th>
<th>movie</th>
<th>music</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vertices</td>
<td>2013</td>
<td>29448</td>
<td>25592</td>
<td>52349</td>
</tr>
<tr>
<td>Edges(K)</td>
<td>-</td>
<td>84.8</td>
<td>435.8</td>
<td>178.0</td>
</tr>
</tbody>
</table>

### Test Data

<table>
<thead>
<tr>
<th>Data Size</th>
<th>user</th>
<th>book</th>
<th>Line vector size: 15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vertices</td>
<td>90</td>
<td>100</td>
<td>Test 100 (user,book) pair</td>
</tr>
</tbody>
</table>
## Experiment Result

<table>
<thead>
<tr>
<th>Meta-path</th>
<th>Support(F-1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>UB (user-book)</td>
<td>0.358</td>
</tr>
<tr>
<td>UUB</td>
<td>0.358</td>
</tr>
<tr>
<td>UBB</td>
<td>0.358</td>
</tr>
<tr>
<td>UBBB</td>
<td>0.358</td>
</tr>
<tr>
<td>UVB (V=movie)</td>
<td>0.358</td>
</tr>
<tr>
<td>USB (S=music)</td>
<td>0.358</td>
</tr>
<tr>
<td>UBUB</td>
<td>0.346</td>
</tr>
<tr>
<td>UUUB</td>
<td>0.358</td>
</tr>
<tr>
<td>UBBB</td>
<td>0.358</td>
</tr>
</tbody>
</table>
Path Predict and Graph Embedding (5 labels)

Increase compared to Random Prediction (%)

- UBUB
- USUB
- UVUB

F1 score
Path Predict and Graph Embedding (2 labels)

Increase compared to Random Prediction (%)

- UBUB
- USUB
- UVUB

F1 score
Challenges

1. Techniques to deal with various ratings
2. Data Size & Computation Time
   a. Local matrix multiplication parallelization using pandas -> numpy API
   b. Matrix partition & GNU parallel & binary local prestorage (NPY)
   c. Sampling
3. Data mismatch after sampling and LINE cutting plane approximation
4. Accuracy of the prediction model (e.g. LR)
5. Hard to improve F1 score in recommendation in general
Comments or Questions?