Modeling User Networks in Recommender Systems

3RD INTERNATIONAL WORKSHOP ON SEMANTIC MEDIA ADAPTATION AND PERSONALIZATION

Dimitrios Vogiatzis,  
NCSR “Demokritos”,  
Athens, Greece  
dimitrv@iit.demokritos.gr

Nicolas Tsapatsoulis,  
Cyprus University of Technology,  
Limassol, Cyprus  
nicolas.tsapatsoulis@cut.ac.cy
Presentation Summary

- Aim and rational
- Power-law & Previous work in the field
- Recommender systems
- User’s (social) network
- Question to explore
- Experimental Setting
- Results
- Conclusions
- Future work
- PServer
Aim and rational

- Discover the nature of social networks formed by recommender systems
  - Discover any laws connecting the ranking of user (based on the number of similarities with other users) with the number of similarities with other users
    - Hypothesis: Power-law holds

- Rational
  - Synthetic data generation for fast proof of concept
  - Identification of influential users or experts in a field
Literature on Power laws: \( P(X=x) = k x^{-a} \)

- **Internet topology (1997-1998)**
  - Power-law: Outdegree of nodes & their rank (Faloutsos et al. 1999)

- **Recommender systems**
  - Power-law: Influential users & their ranking (Rachid et al. 2005)
  - Power-law: Network value of individual users & their ranking
    - network value of customers = marketing value of a customer in terms of other customers that maybe influenced by him (P. Domingos et al. 2001)

- **Observed Elsewhere**
  - Power-laws in: Physics, biology, geology
Recommender systems

- User modeling
- Computation of common user preferences
- *Link* people that have something in common
- In Collaborative Filtering
  - Active user receives recommendation about product
  - Recommendation based on similarity of user to other users (community)
  - Similarity is high if two users have evaluated similarly a number of items in the past
  - Item unknown to the user that belongs to a community will receive recommendation based on the community’s evaluation
Network of users

- Nodes represent users
- If two users are similar \( \rightarrow \) edge connects them
- Similarity \( s \) definition for users: a and b

\[
s := 1 - \frac{H(a, b)}{n}
\]

- \( a = (1, 0, 0, 1, ...), \) movie ratings for user a, e.g. 1: positive 0: negative
- \( b = (1, 0, 1, 0, ...), \) movie ratings for user b
- \( n = \) number of commonly evaluated items
- \( H = \) hamming distance
- \( s \) in \([0, 1]\), higher values \( \rightarrow \) closer similarity

- Degree of a node = \#edges connecting node to other sufficiently similar nodes
Questions to explore

- **Is there a power law distribution observed?**
  - $P(X=x) = k x^{-a}$
  - $\log(p(X=x)) = (-a) \log(x) + \log(k)$
  - $x$: ranking of node’s outdegree
  - $P(X=x)$: node’s outdegree

- **Does the rule hold over time?**
  - As more users are added, or more rankings are added?
Power Law

Power law distribution for various value of \( \alpha \)

- \( \alpha = 0.1 \)
- \( \alpha = 0.3 \)
- \( \alpha = 0.5 \)

Outerdegree of Nodes vs. Ranking (based on outerdegree)
Experimental Setting: Movie Lens Data

Data sets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Users</th>
<th>Movies</th>
<th>Evaluations</th>
<th>Time Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small movie Lens*</td>
<td>943</td>
<td>1,682</td>
<td>100,000 scale 1-5</td>
<td>9/1997-4/1998</td>
</tr>
<tr>
<td>Big movie Lens*</td>
<td>6,040</td>
<td>3,952</td>
<td>1,000,000 scale 1-5</td>
<td>4/2000-2/2003</td>
</tr>
</tbody>
</table>

What do we study

- Degree of node vs ranking of node
- Nodes denote users
- Degree of node = #neighbours
- Evolution of the above over time

Movie Lens data form

<table>
<thead>
<tr>
<th>UserID</th>
<th>MovieID</th>
<th>Rating</th>
<th>TimeStamp</th>
</tr>
</thead>
<tbody>
<tr>
<td>User-1</td>
<td>Movie-1</td>
<td>3/5</td>
<td>1233545</td>
</tr>
<tr>
<td>User-1</td>
<td>Movie-2</td>
<td>4/5</td>
<td>45545666</td>
</tr>
</tbody>
</table>

Data set

- Group Lens Lab, Dept. of CS, Univ. of Minnesota
- Each user has evaluated ≥ 20 movies
- Eval. grade 1,2 → 0, 3,4,5→1
- Split each data set into four time periods (data sorted according to timeStamp)
  - T1 start: 1/4 data
  - T2 start: 2/4 data
  - T3 start: 3/4 data
  - T4 start: end of data
Results summarised

\[ Z = \log_{10}(y) = a \times x + b \]

- Relation holds for both *small & big*
- Relation holds true over time
- \( y \): node degree
- \( x \): ranking of node
- \( a, b \): parameters
  - Depend on data set
  - Change over time altering slope of line \( z \)

A *power law* was **not found**! If it were found it would like: \( \log_{10}(y) = a \times \log_{10}(x) + b \)
### Results summarised (cont.)

<table>
<thead>
<tr>
<th>SMALL</th>
<th>Mean degree</th>
<th># users</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>T-1</td>
<td>5.14</td>
<td>280</td>
<td>8e-3</td>
</tr>
<tr>
<td>T-2</td>
<td>13.72</td>
<td>491</td>
<td>3.7e-3</td>
</tr>
<tr>
<td>T-3</td>
<td>20.81</td>
<td>708</td>
<td>6.1e-3</td>
</tr>
<tr>
<td>T-4</td>
<td>31.86</td>
<td>943</td>
<td>1.4e-2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BIG</th>
<th>Mean degree</th>
<th># users</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>T-1</td>
<td>15.93</td>
<td>1772</td>
<td>4.1e-3</td>
</tr>
<tr>
<td>T-2</td>
<td>36.12</td>
<td>3255</td>
<td>9.1e-3</td>
</tr>
<tr>
<td>T-3</td>
<td>38.58</td>
<td>5140</td>
<td>3.6e-3</td>
</tr>
<tr>
<td>T-4</td>
<td>64.26</td>
<td>6040</td>
<td>2.8e-3</td>
</tr>
</tbody>
</table>

Mean degree: mean number of edges of node (user) stemming out to reach other users

Next Slide plots
Discovered Law (small data set)
Discovered Law (small data set)
Discovered Law (big data set)
Discovered Law (big data set)
Results: Small Data Set
Results: Big Data Set
Conclusions

- **Recommender System Network:**
  - Social networks created by recommender systems do not follow a power law. It seems that an exponential law is more accurate.
  - As the network size increases (as a result of new users and new evaluations) the slope of the exponential function decreases. This means that the biggest the network the more difficult to identify influential users or experts.
  - Modeling of networks feasible.

- **Issues:**
  - Similarity measure
  - Product type
Future work

- Expand work on other data sets
  - We possess a large collection of apparels orders by online customers
  - Customers: gender, age
  - Apparels: size, material, pattern, style, etc...

- Integrate information about demographics of user, to see if we observe the same behaviour
  - Use user stereotypes
  - for that use PServer (see next slide)
PServer

- general-purpose personalization server developed @NCSR “Demokritos” (Skel lab, www.iit.demokritos.gr)
- PServer will be released as open source
- Separates user modeling modules from rest of application at both *logical & physical levels*
  - 2 important concepts in PServer
    - **Attributes**: refer to application independent data, such as user demographics (e.g. age, sex, occupation)
    - **Features**: which are application characteristics that may or may not attract user preference (e.g. Film genres...)

PServer: Logical Level

- **PServer purpose:** user modeling
- **Individual User models:** user specific & made of features & attributes.
  - Attributes have constant values (e.g. age, sex),
  - features are application specific and they assume values depending on user interest (e.g. comedy, thriller, etc.)
- **Stereotypes:** predefined user groups with certain common attributes & features. Stereotypes have features & attributes like user models, but it is not necessary for all stereotypes to have the same number of features and attributes. In the future stereotypes will be automatically derived
- **User & features communities** can be created with machine learning algorithms based on user interaction data (feature values), thus they are data driven
  - For users we would like to discover the ones which are similar with respect to their interest expressed in feature values
  - For features we would like to discover which have the similar values for various users. (i.e. if a feature is equally important as another one, then it can be suggested to a new user
Pserver: user models & stereotypes

Example of user model: many user models fall in a stereotype, with a certain degree

<table>
<thead>
<tr>
<th>ID</th>
<th>age</th>
<th>sex</th>
<th>occupation</th>
<th>Zip-code</th>
<th>Romance</th>
<th>Comedy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234</td>
<td>25</td>
<td>Female</td>
<td>Engineer</td>
<td>55</td>
<td>4/4</td>
<td>…</td>
</tr>
</tbody>
</table>

Example of stereotype, contains ranges of values for some features

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Features’ rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>age</td>
<td>Romance</td>
</tr>
<tr>
<td>young (20-30)</td>
<td>Comedy</td>
</tr>
<tr>
<td>gender</td>
<td>4-5</td>
</tr>
<tr>
<td>Female</td>
<td>4-5</td>
</tr>
</tbody>
</table>

Future work: automatic derivation of stereotypes
PServer: Physical Level

- PServer may reside at a different machine from the recommender application.
- PServer: implemented as a Web server that listens to a dedicated port, all requests have the form of HTTP messages.
- Web browsers can be used as a PServer clients.
- Responses: encoded in XML, & especially made XSL stylesheets allow them to be displayed on web browsers.
- To facilitate applications:
  - Available client-side library of classes is available.
  - Classes can incorporated into the application to handle all low-level communication details.
PServer

- PServer has already been used successfully in a number of European and national projects
  - Xenios/Xenios (Personalised Information presentation in web museums, national)
  - Crossmark (Personalised information extraction from the Web)
  - Servive (SERvice Oriented Intelligent Value Adding Network, FP7)
PServer screen shots
Thank you

Questions?