CONTEXT-RICH RECOMMENDATION: INTEGRATING LINKS, TEXT, AND SPATIO-TEMPORAL DIMENSIONS

Yizhou Sun (University of California, Los Angeles)
Xiang Ren (University of Southern California)
Hongzhi Yin (The University of Queensland, Australia)

August 11, 2017
Outline

• Part I: Introduction and Preliminaries

• Part II: Recommendation in Heterogeneous Information Networks

• Part III: Recommendation in a Text-Rich Setting

• Part IV: Recommendation with Spatio-Temporal Information

• Part V: Research Frontiers and Summary
Traditional View of Recommendation

Avatar
Aliens
Titanic
Revolutionary Road
Recommendation Paradigm

- **Recommendation Paradigm**: The process of recommendation involves user-item feedback, product features, and external knowledge.

**User-Item Feedback**
- **Collaborative Filtering**
  - E.g., K-Nearest Neighbor (Sarwar WWW'01), Matrix Factorization (Hu ICDM'08, Koren IEEE-CS'09), Probabilistic Model (Hofmann SIGIR'03)

**Product Features**
- Content-Based Methods (E.g., Content-Based CF (Antonopoulus, IS'06), External Knowledge CF (Ma WSDM'11))

**External Knowledge**
An Example of Traditional Method: Matrix Factorization

**$R$: Rating Matrix**

\[
\begin{array}{cccccccc}
 & i_1 & i_2 & i_3 & i_4 & i_5 & i_6 & i_7 & i_8 \\
 u_1 & 5 & 2 & 3 & 4 & & & & \\
 u_2 & 4 & 3 & 5 & & & & & \\
 u_3 & 4 & 2 & 2 & 4 & & & & \\
 u_4 & & & & & & & & \\
 u_5 & 5 & 1 & 2 & 4 & 3 & & & \\
 u_6 & 4 & 3 & 2 & 4 & 3 & 5 & & \\
\end{array}
\]

\[
U = \begin{bmatrix}
1.55 & 1.22 & 0.37 & 0.81 & 0.62 & -0.01 \\
0.36 & 0.91 & 1.21 & 0.39 & 1.10 & 0.25 \\
0.59 & 0.20 & 0.14 & 0.83 & 0.27 & 1.51 \\
0.39 & 1.33 & -0.43 & 0.70 & -0.90 & 0.68 \\
1.05 & 0.11 & 0.17 & 1.18 & 1.81 & 0.40 \\
\end{bmatrix}
\]

**$\hat{R}$: Estimated Rating Matrix**

\[
\begin{array}{cccccccc}
 & i_1 & i_2 & i_3 & i_4 & i_5 & i_6 & i_7 & i_8 \\
 u_1 & 5 & 2 & 2.5 & 3 & 4.8 & 4 & 2.2 & 4.8 \\
 u_2 & 4 & 3 & 2.4 & 2.9 & 5 & 4.1 & 2.6 & 4.7 \\
 u_3 & 4 & 1.7 & 2 & 3.2 & 3.9 & 3.0 & 2 & 4 \\
 u_4 & 4.8 & 2.1 & 2.7 & 2.6 & 4.7 & 3.8 & 2.4 & 4.9 \\
 u_5 & 5 & 1 & 2 & 3.4 & 4 & 3 & 1.5 & 4.6 \\
 u_6 & 4 & 3 & 2.9 & 2 & 4 & 3.4 & 3 & 5 \\
\end{array}
\]

\[
V = \begin{bmatrix}
1.00 & -0.05 & -0.24 & 0.26 & 1.28 & 0.54 & -0.31 & 0.52 \\
0.19 & -0.86 & -0.72 & 0.05 & 0.68 & 0.02 & -0.61 & 0.70 \\
0.49 & 0.09 & -0.05 & -0.62 & 0.12 & 0.08 & 0.02 & 1.60 \\
-0.40 & 0.70 & 0.27 & -0.27 & 0.99 & 0.44 & 0.39 & 0.74 \\
1.49 & -1.00 & 0.06 & 0.05 & 0.23 & 0.01 & -0.36 & 0.80 \\
\end{bmatrix}
\]
Challenges

• How to address the data sparsity and cold start issues?

• How to integrate content information, such as text, into the recommendation?

• How to do spatio-temporal recommendation?
Solution: A Heterogeneous Information Network
View of Recommendation

Avatar
Aliens
Titanic
Revolutionary Road

Zoe Saldana
Adventure
James Cameron
Leonardo DiCaprio
Kate Winslet

Romance
What Are Information Networks?

• A network where each **node** represents an **entity** (e.g., user in a social network) and each **link** (e.g., friendship) a **relationship** between entities.
  
  • Nodes/links may have attributes, labels, and weights.
  
  • Links may carry rich semantic information.
Heterogeneous Information Networks

1. Multiple entity types and link types
2. New problems are emerging in heterogeneous networks!
We are living in a connected world!
Even in Biomedical Domain

- Gene
- Microbe
- Disease
- Symptom
- Patient
- Drug
- Side Effect
- Compound

- Gene carriedBy Microbe
- Disease cause Symptom
- Patient has Symptom and Disease
- Drug treat Patient
- Drug has Compound
- Compound similarTo Drug
Outline

• Part I: Introduction and Preliminaries

• Part II: Recommendation in Heterogeneous Information Networks

• Part III: Recommendation in a Text-Rich Setting

• Part IV: Recommendation with Spatio-Temporal Information

• Part V: Research Frontiers and Summary
Part II: Recommendation in Heterogeneous Information Networks

- Hybrid Collaborative Filtering with Information Networks
- Graph Regularization for Recommendation
- Network Embedding-based Entity Recommendation
- Neural Network-based Collaborative Filtering
Recommendation Paradigm

**User-Item Feedback**

**Product Features**

**External Knowledge**

**Collaborative Filtering**
- E.g., K-Nearest Neighbor (Sarwar WWW'01), Matrix Factorization (Hu ICDM'08, Koren IEEE-CS'09), Probabilistic Model (Hofmann SIGIR'03)

**Content-Based Methods**
- E.g., (Balabanovic Comm. ACM'97, Zhang SIGIR'02)

**Hybrid Methods**
- E.g., Content-Based CF (Antonopoulus, IS'06), External Knowledge CF (Ma WSDM'11)

**Recommendation System**

**Feedback**

**User**

**Recommendation**
Problem Definition

hybrid collaborative filtering with information networks
Hybrid Collaborative Filtering with Networks

• Utilizing network relationship information can enhance the recommendation quality

• However, most of the previous studies only use single type of relationship between users or items (e.g., social network Ma, WSDM’11, trust relationship Ester, KDD’10, service membership Yuan, RecSys’11)
The Heterogeneous Information Network View of Recommender System
**Relationship Heterogeneity Alleviates Data Sparsity**

**Collaborative filtering methods suffer from data sparsity issue**

A small number of users and items have a large number of ratings

Most users and items have a small number of ratings

- Heterogeneous relationships complement each other
- Users and items with limited feedback can be connected to the network by different types of paths
  - Connect new users or items *(cold start)* in the information network
Different users may have different behaviors or preferences

Two levels of personalization

Data level
• Most recommendation methods use one model for all users and rely on personal feedback to achieve personalization

Model level
• With different entity relationships, we can learn personalized models for different users to further distinguish their differences
Preference Propagation-Based Latent Features

- Generate $L$ different \textbf{meta-path} (path types) connecting users and items
- Propagate user implicit feedback along each meta-path
- Calculate latent features for users and items for each meta-path with \textbf{NMF} related method
**Observation 1:** Different meta-paths may have different importance

Global Recommendation Model

\[
\hat{r}(u_i, e_j) = \sum_{q=1}^{L} \theta_q \cdot \hat{U}_i(q) \hat{V}_j(q)^T
\]  
(1)

**Observation 2:** Different users may require different models

Personalized Recommendation Model

\[
\hat{r}_p(u_i, e_j) = \sum_{k=1}^{c} \text{sim}(C_k, u_i) \sum_{q=1}^{L} \theta_q^{\{k\}} \cdot \hat{U}_i(q) \hat{V}_j(q)^T
\]  
(2)

where \( \text{sim}(C_k, u_i) \) is the user-cluster similarity.
Parameter Estimation

- Bayesian personalized ranking \( \text{(Rendle UAI’09)} \)
- Objective function

\[
\min_\Theta - \sum_{u_i \in U} \sum_{(e_a > e_b) \in R_i} \ln \sigma(\hat{r}(u_i, e_a) - \hat{r}(u_i, e_b)) + \frac{\lambda}{2} \|\Theta\|_2^2
\]

(3)

Soft cluster users with NMF + k-means

For each user cluster, learn one model with Eq. (3)

Generate personalized model for each user on the fly with Eq. (2)

Learning Personalized Recommendation Model
Experiment Setup

• **Datasets**

<table>
<thead>
<tr>
<th>Name</th>
<th>#Items</th>
<th>#Users</th>
<th>#Ratings</th>
<th>#Entities</th>
<th>#Links</th>
</tr>
</thead>
<tbody>
<tr>
<td>IM100K</td>
<td>943</td>
<td>1360</td>
<td>89,626</td>
<td>60,905</td>
<td>146,013</td>
</tr>
<tr>
<td>Yelp</td>
<td>11,537</td>
<td>43,873</td>
<td>229,907</td>
<td>285,317</td>
<td>570,634</td>
</tr>
</tbody>
</table>

• **Comparison methods:**
  - **Popularity**: recommend the most popular items to users
  - **Co-click**: conditional probabilities between items
  - **NMF**: non-negative matrix factorization on user feedback
  - **Hybrid-SVM**: use Rank-SVM with plain features (utilize both user feedback and information network)
## Performance Comparison

<table>
<thead>
<tr>
<th>Method</th>
<th>IM100K</th>
<th></th>
<th></th>
<th></th>
<th>Yelp</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Prec1</td>
<td>Prec5</td>
<td>Prec10</td>
<td>MRR</td>
<td>Prec1</td>
<td>Prec5</td>
<td>Prec10</td>
<td>MRR</td>
</tr>
<tr>
<td>Popularity</td>
<td>0.0731</td>
<td>0.0513</td>
<td>0.0489</td>
<td>0.1923</td>
<td>0.00747</td>
<td>0.00825</td>
<td>0.00780</td>
<td>0.0228</td>
</tr>
<tr>
<td>Co-Click</td>
<td>0.0668</td>
<td>0.0558</td>
<td>0.0538</td>
<td>0.2041</td>
<td>0.0147</td>
<td>0.0126</td>
<td>0.01132</td>
<td>0.0371</td>
</tr>
<tr>
<td>NMF</td>
<td>0.2064</td>
<td>0.1661</td>
<td>0.1491</td>
<td>0.4938</td>
<td>0.0162</td>
<td>0.0131</td>
<td>0.0110</td>
<td>0.0382</td>
</tr>
<tr>
<td>Hybrid-SVM</td>
<td>0.2087</td>
<td>0.1441</td>
<td>0.1241</td>
<td>0.4493</td>
<td>0.0122</td>
<td>0.0121</td>
<td>0.0110</td>
<td>0.0337</td>
</tr>
<tr>
<td>HeteRec-g</td>
<td>0.2094</td>
<td>0.1791</td>
<td>0.1614</td>
<td>0.5249</td>
<td>0.0165</td>
<td>0.0144</td>
<td>0.0129</td>
<td>0.0422</td>
</tr>
<tr>
<td>HeteRec-p</td>
<td>0.2121</td>
<td>0.1932</td>
<td>0.1681</td>
<td>0.5530</td>
<td>0.0213</td>
<td>0.0171</td>
<td>0.0150</td>
<td>0.0513</td>
</tr>
</tbody>
</table>

HeteRec personalized recommendation (HeteRec-p) provides the best recommendation results
Performance under Different Scenarios

HeteRec–p consistently outperform other methods in different scenarios better recommendation results if users provide more feedback better recommendation for users who like less popular items
Part II: Recommendation in Heterogeneous Information Networks

- Hybrid Collaborative Filtering with Information Networks
- Graph Regularization for Recommendation
- Network Embedding-based Entity Recommendation
- Neural Network-based Collaborative Filtering
From Graph Regularization Point of View

• Why additional links help?
  • They define new similarity metrics between users or items.

• How to integrate this assumption into recommendation?
  • Use graph regularization to force two entities to be similar in latent space, if they are similar in graph

• The original form of graph regularization
  \[ \frac{1}{2} \sum w_{ij} (f_i - f_j)^2 = f'Lf \]
  • \( w_{ij} \): similarity of node i and j
  • \( f_i \): some latent representation for node i
  • \( L \): Laplacian matrix of \( W \), i.e., \( L = D - W \),
    • where \( D \) is a diagonal matrix and \( D_{ii} = \sum_j w_{ij} \)
Recommender Systems with Social Regularization
[Ma et al., WSDM’11]

• Input: Social Relation + Rating Matrix

(a) Real World Social Recommendation  (b) Social Network  (c) User-Item Rating Matrix

<table>
<thead>
<tr>
<th></th>
<th>$v_1$</th>
<th>$v_2$</th>
<th>$v_3$</th>
<th>$v_4$</th>
<th>$v_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>1</td>
<td></td>
<td></td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>$u_2$</td>
<td></td>
<td>3</td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>$u_3$</td>
<td></td>
<td></td>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$u_4$</td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>$u_5$</td>
<td></td>
<td>2</td>
<td>5</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>
Two Regularization Forms

- **Model 1: Average-based Regularization**
  - We are similar to the average of our friends

\[
\min_{U,V} \mathcal{L}_1(R, U, V) = \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} I_{ij} (R_{ij} - U_i^T V_j)^2 \\
+ \frac{\alpha}{2} \sum_{i=1}^{m} \frac{1}{|\mathcal{F}^+(i)|} \sum_{f \in \mathcal{F}^+(i)} U_f \|F\|^2_F \\
+ \frac{\lambda_1}{2} \|U\|^2_F + \frac{\lambda_2}{2} \|V\|^2_F. \tag{5}
\]

- **Model 2: Individual-based Regularization**
  - We are similar to each of our friends

\[
\min_{U,V} \mathcal{L}_2(R, U, V) = \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} I_{ij} (R_{ij} - U_i^T V_j)^2 \\
+ \frac{\beta}{2} \sum_{i=1}^{m} \sum_{f \in \mathcal{F}^+(i)} \text{Sim}(i, f) \|U_i - U_f\|^2_F \\
+ \lambda_1 \|U\|^2_F + \lambda_2 \|V\|^2_F. \tag{11}
\]

Similarity can be propagated via friends: transitivity!
How to compute similarity between two users?

- **Cosine similarity (VSS)**

  \[
  Sim(i, f) = \frac{\sum_{j \in I(i) \cap I(f)} R_{ij} \cdot R_{fj}}{\sqrt{\sum_{j \in I(i) \cap I(f)} R_{ij}^2} \cdot \sqrt{\sum_{j \in I(i) \cap I(f)} R_{fj}^2}}
  \]

- **Pearson correlation coefficient (PCC)**

  \[
  Sim(i, f) = \frac{\sum_{j \in I(i) \cap I(f)} (R_{ij} - \overline{R}_i) \cdot (R_{fj} - \overline{R}_f)}{\sqrt{\sum_{j \in I(i) \cap I(f)} (R_{ij} - \overline{R}_i)^2} \cdot \sqrt{\sum_{j \in I(i) \cap I(f)} (R_{fj} - \overline{R}_f)^2}}
  \]

  (14)
# Results

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Training</th>
<th>Metrics</th>
<th>UserMean</th>
<th>ItemMean</th>
<th>NMF</th>
<th>PMF</th>
<th>RSTE</th>
<th>SR1_vss</th>
<th>SR1_pcc</th>
<th>SR2_vss</th>
<th>SR2_pcc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Douban</td>
<td>80%</td>
<td>MAE</td>
<td>0.6809</td>
<td>0.6288</td>
<td>0.5732</td>
<td>0.5693</td>
<td>0.5643</td>
<td>0.5579</td>
<td>0.5576</td>
<td>0.5548</td>
<td><strong>0.5543</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Improve</td>
<td>18.59%</td>
<td>11.85%</td>
<td>3.30%</td>
<td>2.63%</td>
<td>1.77%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>RMSE</td>
<td>0.8450</td>
<td>0.7898</td>
<td>0.7225</td>
<td>0.7200</td>
<td>0.7144</td>
<td>0.7026</td>
<td>0.7022</td>
<td>0.6992</td>
<td><strong>0.6988</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Improve</td>
<td>17.59%</td>
<td>11.52%</td>
<td>3.28%</td>
<td>2.94%</td>
<td>2.18%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>60%</td>
<td>MAE</td>
<td>0.6823</td>
<td>0.6300</td>
<td>0.5768</td>
<td>0.5737</td>
<td>0.5698</td>
<td>0.5627</td>
<td>0.5623</td>
<td>0.5597</td>
<td><strong>0.5593</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Improve</td>
<td>18.02%</td>
<td>11.22%</td>
<td>3.03%</td>
<td>2.51%</td>
<td>1.84%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>RMSE</td>
<td>0.8505</td>
<td>0.7926</td>
<td>0.7351</td>
<td>0.7290</td>
<td>0.7207</td>
<td>0.7081</td>
<td>0.7078</td>
<td>0.7046</td>
<td><strong>0.7042</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Improve</td>
<td>17.20%</td>
<td>11.15%</td>
<td>4.20%</td>
<td>3.40%</td>
<td>2.29%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>40%</td>
<td>MAE</td>
<td>0.6854</td>
<td>0.6317</td>
<td>0.5899</td>
<td>0.5868</td>
<td>0.5767</td>
<td>0.5706</td>
<td>0.5702</td>
<td>0.5690</td>
<td><strong>0.5685</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Improve</td>
<td>17.06%</td>
<td>10.00%</td>
<td>3.63%</td>
<td>3.12%</td>
<td>1.42%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>RMSE</td>
<td>0.8567</td>
<td>0.7971</td>
<td>0.7482</td>
<td>0.7411</td>
<td>0.7295</td>
<td>0.7172</td>
<td>0.7169</td>
<td>0.7129</td>
<td><strong>0.7125</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Improve</td>
<td>16.83%</td>
<td>10.61%</td>
<td>4.77%</td>
<td>3.86%</td>
<td>2.33%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Epinions</td>
<td>90%</td>
<td>MAE</td>
<td>0.9134</td>
<td>0.9768</td>
<td>0.8712</td>
<td>0.865</td>
<td>0.8367</td>
<td>0.8290</td>
<td>0.8287</td>
<td>0.8258</td>
<td><strong>0.8256</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Improve</td>
<td>9.61%</td>
<td>15.48%</td>
<td>5.23%</td>
<td>4.57%</td>
<td>1.33%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>RMSE</td>
<td>1.1688</td>
<td>1.2375</td>
<td>1.1621</td>
<td>1.1544</td>
<td>1.1094</td>
<td>1.0792</td>
<td>1.0790</td>
<td>1.0744</td>
<td><strong>1.0739</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Improve</td>
<td>8.12%</td>
<td>13.22%</td>
<td>7.59%</td>
<td>6.97%</td>
<td>3.20%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>80%</td>
<td>MAE</td>
<td>0.9285</td>
<td>0.9913</td>
<td>0.8951</td>
<td>0.8886</td>
<td>0.8537</td>
<td>0.8493</td>
<td>0.8491</td>
<td>0.8447</td>
<td><strong>0.8443</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Improve</td>
<td>9.07%</td>
<td>14.83%</td>
<td>5.68%</td>
<td>4.99%</td>
<td>1.10%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>RMSE</td>
<td>1.1817</td>
<td>1.2584</td>
<td>1.1832</td>
<td>1.1760</td>
<td>1.1256</td>
<td>1.1016</td>
<td>1.1013</td>
<td>1.0958</td>
<td><strong>1.0954</strong></td>
</tr>
</tbody>
</table>
Meta-Path-based Regularization [Yu et al., IJCAI-HINA’13]

• What if it is more than one type of relation?

<table>
<thead>
<tr>
<th></th>
<th>E1</th>
<th>e2</th>
<th>...</th>
<th>em</th>
</tr>
</thead>
<tbody>
<tr>
<td>u1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>u2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>...</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>un</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Solution:

• Use meta-path to generate similarity relation between items, e.g., movie-director-movie
• Learn the importance score for each meta-path

Rating Data

Heterogeneous Information Network
Notations

• We have $n$ users and $m$ items.
  - $\mathcal{U} = \{u_1, ..., u_n\} \quad \mathcal{I} = \{e_1, ..., e_m\}$

• By computing similarity scores of all item pairs along certain meta-path, we can get a similarity matrix
  - $S^{(l)} \in \mathbb{R}^{n \times n}$

• With $L$ different meta-paths, we can calculate $L$ similarity matrices as
  - $S^{(1)}, S^{(2)}, ..., S^{(L)}$
Objective Function

\[
\min_{U, V, \theta} \left\| Y \odot (R - U V^T) \right\|_F^2 + \lambda_0 (\|U\|_F^2 + \|V\|_F^2) + \frac{\lambda_1}{2} \sum_{i,j} \sum_{l=1}^L \theta_l S_{ij}^{(l)} \left\| V_i - V_j \right\|_2^2 + \lambda_2 \left\| \theta \right\|_2^2,
\]

Regularization on \( U \) and \( V \) product

\[
\frac{\lambda_1}{2} \sum_{i,j} \sum_{l=1}^L \theta_l S_{ij}^{(l)} \left\| V_i - V_j \right\|_2^2
\]

Similar items measured from HIN should have similar low-rank representations

\[
s.t. \quad U \geq 0, \quad V \geq 0, \quad \theta \geq 0, \quad \text{and} \quad \sum_{l=1}^L \theta_l = 1,
\]

Regularization on \( \theta \), which is the importance score for each meta-path
Equivalent Objective Function Using Graph Laplacian

\[ D_{ii}^{(l)} = \sum_{j=1}^{n} S_{ij}^{(l)} \quad L^{(l)} = D^{(l)} - S^{(l)} \]

\[
\min_{U,V,\theta} \quad \| Y \odot (R - UV^T) \|_F^2 + \lambda_0 (\| U \|_F^2 + \| V \|_F^2) + \\
\lambda_1 \cdot \text{Tr} \left( V^T \left( \sum_{l=1}^{L} \theta_l L^{(l)} \right) V \right) + \lambda_2 \| \theta \|_F^2,
\]

Similar items measured from HIN should have similar low-rank representations

s.t. \[ U \geq 0, \quad V \geq 0, \quad \theta \geq 0, \quad \text{and} \quad \sum_{l=1}^{L} \theta_l = 1. \]
Dataset

- We combine IMDb + MovieLens100K

<table>
<thead>
<tr>
<th>Name</th>
<th>#Items</th>
<th>#Users</th>
<th>#Ratings</th>
<th>#Entities</th>
<th>#Links</th>
</tr>
</thead>
<tbody>
<tr>
<td>IM100K</td>
<td>943</td>
<td>1360</td>
<td>89,626</td>
<td>60,905</td>
<td>146,013</td>
</tr>
</tbody>
</table>

(a) Datasets Description

We random sample training datasets of different sizes (0.4, 0.6, and 0.8)
## Results

<table>
<thead>
<tr>
<th>Metric</th>
<th>MAE</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>40%</td>
<td>60%</td>
</tr>
<tr>
<td>Training Size</td>
<td>0.8400</td>
<td>0.8409</td>
</tr>
<tr>
<td>UserMean</td>
<td>0.8167</td>
<td>0.8237</td>
</tr>
<tr>
<td>ItemMean</td>
<td>2.1944</td>
<td>2.1862</td>
</tr>
<tr>
<td>NMF (d=40)</td>
<td>0.7919</td>
<td>0.7879</td>
</tr>
<tr>
<td>WNMF (d=10)</td>
<td>0.7917</td>
<td>0.7875</td>
</tr>
<tr>
<td>WNMF (d=20)</td>
<td>0.7886</td>
<td>0.7833</td>
</tr>
<tr>
<td>WNMF (d=40)</td>
<td>0.7838</td>
<td>0.7800</td>
</tr>
<tr>
<td>Hete-MF (d=10)</td>
<td>0.7818</td>
<td>0.7802</td>
</tr>
<tr>
<td>Hete-MF (d=20)</td>
<td>0.7780</td>
<td>0.7772</td>
</tr>
<tr>
<td>Hete-MF (d=40)</td>
<td>0.7780</td>
<td>0.7772</td>
</tr>
</tbody>
</table>
Part II: Recommendation in Heterogeneous Information Networks

- Hybrid Collaborative Filtering with Information Networks
- Graph Regularization for Recommendation
- Network Embedding-based Entity Recommendation
- Neural Network-based Collaborative Filtering
Task-Guided Meta-Path Augmented Embedding
[Chen et al., WSDM’17]

• **Given an anonymized paper, with**
  - Venue (e.g., WSDM)
  - Year (e.g., 2017)
  - Keywords (e.g., “heterogeneous network embedding”)
  - References (e.g., [Chen et al., IJCAI’16])

• Can we predict its authors?
Challenge 1: Task Guided Embedding

Embedding for Movie Recommendation  Embedding for Voting Prediction
Challenge 2: Heterogeneous Network Embedding

• How to utilize links belonging to different types with different semantic meanings?
Our Solution

- **Task-guided and path-augmented embedding: A Semi-Supervised framework**
  - Task-guided embedding takes care of supervised labels
    - E.g., Author “Ting Chen” should be close to Keyword “Heterogeneous network embedding”
  - Path-augmented embedding takes care of the global structure of networks (Path-augmented network regularization)
    - E.g., Keyword “heterogeneous network embedding” should be close to Keyword “node representation”
      - meta-path: Keyword-Paper->Paper-Keyword
The Combined Model

• Joint training of two types of embedding

• Path selection is performed to pick most informative meta-paths for network embedding.
Component 1: Task-Guided Embedding

- The embedding architecture for author identification
Formally

• Consider the ego-network of $p$: $X_p = (X_p^1, X_p^2, \ldots, X_p^T)$,
  - $T$: the number of types of nodes associated with paper type
  - $X_p^t$: the set of nodes with type $t$ associated with paper $p$

• $u_a$: embedding of author $a$

• $u_n$: embedding of node $n$

• $V_p$: embedding of paper $p$
  - Weighted average of all the neighbors

• The score function between $p$ and $a$ is defined as:
  \[
  f(p, a) = u_a^T V_p = u_a^T \left( \sum_t w_t V_p^{(t)} \right) \\
  = u_a^T \left( \sum_t w_t \sum_{n \in X_p^{(t)}} u_n / |X_p^{(t)}| \right)
  \]
Ranking-based Objective

• Given a paper $p$, author $a$ that is an author of $p$, and author $b$ that is not an author of $p$
  • $f(p, a) > f(p, b)$

• A hinge loss function with margin is used as objective function
  • $\max \left( 0, f(p, b) - f(p, a) + \xi \right)$
Component 2: Path-Augmented Embedding

- **Limitations of task-guided embedding**
  - Supervised labels expensive to obtain
  - The rich structure information of heterogeneous information networks is not fully explored

- **Path-Augmented Embedding**
  - Prepare meta-paths that are potentially related to the task
    - author-paper-author
    - author-paper->paper
    - author-paper
  - Apply general purpose embedding
Formally

• **For each meta-path-based relation**
  - Define the probability of reaching node $j$ from node $i$ via meta-path $r$ via their embeddings

$$P(j|i; r) = \frac{\exp(u_i^T u_j)}{\sum_{j' \in DST(r)} \exp(u_i^T u_{j'})}$$

  - Use negative sampling to approximate the distribution
  - Extend LINE [Tang et al., 2015]

• The goal is to maximize the likelihood to observing all the paths under each meta-path
The Joint Model

Objective function

\[ L = (1 - \omega)L_{\text{task-specific}} + \omega L_{\text{network-general}} + \Omega(M) \]

\[ = (1 - \omega)\mathbb{E}_{(p,a,b)} \left[ \max \left( 0, f(p, b) - f(p, a) + \xi \right) \right] \]

\[ + \omega \mathbb{E}_{(r,i,j)} \left[ -\log \hat{P}(j|i; r) \right] + \lambda \sum_i \| u_i \|^2_2 \]
How to select meta-paths?

• **A greedy strategy is used to select meta-paths**
  - Step 1: Rank single meta-path according to their performance
  - Step 2: Greedily add the current best meta-path into current pool, stop until the performance deteriorates

• **Different meta-paths will be selected for different tasks**

![Graph showing MAP@3 for different paths]

- **No path/network**
- **Single path network**

![Bar chart showing MAP@3 for different greedy additive paths]

- A2P
- +A2W
- +P1A
- +P1P
- +A2V
- +P2V
- +P2W
- +P1W +W2W
Experiments

**Dataset:**

- AMiner Citation data set.
- Papers before 2012 are used in training, and papers on and after 2012 are used as test.

### Table 1: Node statistics

<table>
<thead>
<tr>
<th></th>
<th>Paper</th>
<th>Author</th>
<th>Keyword</th>
<th>Venue</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>1.6M</td>
<td>1M</td>
<td>4M</td>
<td>7K</td>
<td>60</td>
</tr>
<tr>
<td>Test</td>
<td>34K</td>
<td>62K</td>
<td>42K</td>
<td>1K</td>
<td>2</td>
</tr>
</tbody>
</table>

### Table 3: Length-2 link statistics

<table>
<thead>
<tr>
<th></th>
<th>A-P-A</th>
<th>A-P-P</th>
<th>A-P-V</th>
<th>A-P-W</th>
<th>A-P-Y</th>
<th>P-P-V</th>
<th>P-P-W</th>
<th>V-P-W</th>
<th>W-P-W</th>
<th>Y-P-W</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>17M</td>
<td>18M</td>
<td>4M</td>
<td>38M</td>
<td>4M</td>
<td>3M</td>
<td>27M</td>
<td>12M</td>
<td>118M</td>
<td>12M</td>
</tr>
</tbody>
</table>
Baselines

- Supervised feature-based baselines (i.e. LR, SVM, RF, LambdaMart).
  - Manually crafted features
- Task-specific embedding.
- Network-general embedding.
- Pre-training + Task-specific embedding.
  - Take general embedding as initialization of task-specific embedding
Comparison

- Easy task: choose author candidate as true authors + negative authors

Table 5: Author identification performance comparison.

<table>
<thead>
<tr>
<th>Models</th>
<th>MAP@3</th>
<th>MAP@10</th>
<th>Recall@3</th>
<th>Recall@10</th>
</tr>
</thead>
<tbody>
<tr>
<td>LR</td>
<td>0.7289</td>
<td>0.7321</td>
<td>0.6721</td>
<td>0.8209</td>
</tr>
<tr>
<td>SVM</td>
<td>0.7332</td>
<td>0.7365</td>
<td>0.6748</td>
<td>0.8267</td>
</tr>
<tr>
<td>RF</td>
<td>0.7509</td>
<td>0.7543</td>
<td>0.6921</td>
<td>0.8381</td>
</tr>
<tr>
<td>LambdaMart</td>
<td>0.7511</td>
<td>0.7420</td>
<td>0.6869</td>
<td>0.8026</td>
</tr>
<tr>
<td>Task-specific</td>
<td>0.6876</td>
<td>0.7088</td>
<td>0.6523</td>
<td>0.8298</td>
</tr>
<tr>
<td>Pre-train+Task.</td>
<td>0.7722</td>
<td>0.7962</td>
<td>0.7234</td>
<td>0.9014</td>
</tr>
<tr>
<td>Network-general</td>
<td>0.7563</td>
<td>0.7817</td>
<td>0.7105</td>
<td>0.8903</td>
</tr>
<tr>
<td>Combined</td>
<td>0.8113</td>
<td>0.8309</td>
<td>0.7548</td>
<td>0.9215</td>
</tr>
</tbody>
</table>
Which meta-paths are selected?

- **A-P->P**: author *write* paper *cite* paper
- **A-P-W**: author *write* paper *contain* keyword
- **P-A**: paper *written-by* author
Performance over Different Groups of Authors

Significant enhancement for long-tail authors!
The Real Game

- Treat all the authors as candidates
- Future work: full text analysis
Case Study

Top ranked authors for **Keyword “variational inference”**

<table>
<thead>
<tr>
<th>Task-specific</th>
<th>Network-general</th>
<th>Combined</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chong Wang</td>
<td>Yee Whye Teh</td>
<td>Michael I. Jordan</td>
</tr>
<tr>
<td>Qiang Liu</td>
<td>Mohammad E. Khan</td>
<td>Yee Whye Teh</td>
</tr>
<tr>
<td>Sheng Gao</td>
<td>Edward Challis</td>
<td>Zoubin Ghahramani</td>
</tr>
<tr>
<td>Song Li</td>
<td>Ruslan Salakhutdinov</td>
<td>John William Paisley</td>
</tr>
<tr>
<td>Donglai Zhu</td>
<td>Michael I. Jordan</td>
<td>David M. Blei</td>
</tr>
<tr>
<td>Neil D. Lawrence</td>
<td>Zoubin Ghahramani</td>
<td>Max Welling</td>
</tr>
<tr>
<td>Sotirios Chatzis</td>
<td>Matthias Seeger</td>
<td>Alexander T. Ihler</td>
</tr>
<tr>
<td>Si Wu</td>
<td>David B. Dunson</td>
<td>Eric P. Xing</td>
</tr>
<tr>
<td>Huan Wang</td>
<td>Dae Il Kim</td>
<td>Ryan Prescott Adams</td>
</tr>
<tr>
<td>Weimin Liu</td>
<td>Pradeep D. Ravikumar</td>
<td>Thomas L. Griffiths</td>
</tr>
</tbody>
</table>
Part II: Recommendation in Heterogeneous Information Networks

• Hybrid Collaborative Filtering with Information Networks
• Graph Regularization for Recommendation
• Network Embedding-based Entity Recommendation
• Neural Network-based Collaborative Filtering
Application: News Recommendation

- Chen et al., “On Sampling Strategies for Neural Network-based Collaborative Filtering,” KDD’17

- Use the news recommendation as a running example

Yahoo news feed

**Politics**
President Obama Having Copies Of FEMA Camp Keys Made For Hillary
WASHINGTON, D.C. -- Reporters in the nation's capital recently stumbled upon President Barack Hussein Obama (D-Kenya) at the Home Depot...
The Huffington Post

**Politics**
These tweets reveal why it's so hard for conservatives to oppose Trump
For a lot of liberals, the refusal of major Republicans like House Speaker Paul Ryan and even frequent Trump critics like Sen. John McCain or the Bush...
Vox

**Politics**
The case for Trump?
I have been and remain a never-Trumper. Nevertheless, I read conservative (and even the rare libertarian) defenses of Trump that I see on social media...
The Washington Post

**U.S.**
Mother Denied Parole After Broiling Her 14-Month-Old Daughter Alive in a 600-Degree Oven
An Alabama mother sentenced to 25 years in jail for putting her 14-month-old daughter in a 600-degree oven has officially been denied parole this...

Which news will a user click?

Goal: learning to predict a user’s interests on items (news, articles) based on their text content.
Solution

- Neural Network-based Collaborative Filtering

Subsumes several existing work, e.g., Bansal et al., RecSys’16, Van den Oord et al., NIPS’13
More Generally: Functional Embedding

$\mathbf{r}(u, v) = f(\mathbf{x}_u)^T g(\mathbf{x}_v)$

$\mathbf{x}_u, \mathbf{x}_v \in \mathbb{R}^d$

Goal: Minimize the loss function between predicted and observed rating or ranking.

Figure 1: Computational flow of the proposed framework.
Challenge

- Computational cost is very heavy when embedding functions are complex multi-layer non-linear transformations, such as RNN and CNN.

Figure 2: Model training time per epoch with different types of item functions (in log-scale).
Three Types of Computations

Major Computation Cost Breakdown

(both forward/backward)

User function computation

Item function computation

Interaction function (dot product) computation

\[ t_f \]

\[ t_g \]

\[ t_i \]

10

100

1

Very rough order of magnitude estimate of time units (depending on specific configurations)
The Cost Model in a Mini-Batch

• Assume in a Mini-Batch, we have
  • # users
  • # items
  • # user–item interactions

• Computation cost:
  • $t_f \times \#users + t_g \times \#items + t_i \times \#interactions$

Most expensive!
Solution to Speedup Computation

• Design sampling strategies that can share the computational costs on the node type that are expensive

  • Data to sample here: Links between users and items
  • Major computational costs: on nodes, esp. on items that involves rich text

Pointwise Loss

Pairwise Loss
Existing Sampling Strategies

- **Negative Sampling**: b positive links and k negative links for each positive link
  - In each mini-batch: no items are shared!

- # users: b
- # items: \((1 + k)b\)
- # interactions: \((1 + k)b\)

- Main cost: \(t_g \ast (1 + k)b\)
Proposed Strategy 1: Stratified Sampling

- Share items: $b$ positive links and $k$ negative links for each positive link; # of positive link per item: $s$

(a) Negative
- # users: $b$
- # items: $(1 + k)b$
- # interactions: $(1 + k)b$

(b) Stratified (by Items)
- # users: $(1 + k)b$
- # items: $b/s$
- # interactions: $(1 + k)b$

Main cost: $t_g \times (1 + k)b$

Main cost: $t_g \times b/s$
Proposed Strategy 2: Negative Sharing

- Treat all the non-links as negative links, again share items: still b positive links

(a) Negative

- # users: \( b \)
- # items: \((1 + k)b\)
- # interactions: \((1 + k)b\)

- Main cost: \( t_g \cdot (1 + k)b \)

(c) Negative Sharing

- # users: \( b \)
- # items: \( b \)
- # interactions: \( b^2 \)

- Main cost: \( t_g \cdot b \)
Combine Two Strategies

- **Stratified sampling only**: Cannot deal with ranking-based loss functions
- **Negative sharing only**: Too many negative links used, diminishing return

(b) Stratified (by Items)  
(c) Negative Sharing

(d) Stratified with N.S.
- # users: $b$
- # items: $b/s$
- # interactions: $b \times b/s$

Main cost: $t_g \times b/s$
## Cost Summary

<table>
<thead>
<tr>
<th>Sampling</th>
<th># pos. links</th>
<th># neg. links</th>
<th>$t_f$</th>
<th>$t_g$</th>
<th>$t_i$</th>
<th>pointwise</th>
<th>pairwise</th>
</tr>
</thead>
<tbody>
<tr>
<td>IID [3]</td>
<td>$b$</td>
<td>$bk$</td>
<td>$b(1 + k)$</td>
<td>$b(1 + k)$</td>
<td>$(1 + k)$ vec</td>
<td>✓</td>
<td>×</td>
</tr>
<tr>
<td>Negative [1, 21, 29]</td>
<td>$b$</td>
<td>$bk$</td>
<td>$b(1 + k)$</td>
<td>$b(1 + k)$</td>
<td>$(1 + k)$ vec</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Stratified (by Items)</td>
<td>$b$</td>
<td>$bk$</td>
<td>$b(1 + k)$</td>
<td>$\frac{b}{s}$</td>
<td>$b(1 + k)$ vec</td>
<td>✓</td>
<td>×</td>
</tr>
<tr>
<td>Negative Sharing</td>
<td>$b$</td>
<td>$b(b - 1)$</td>
<td>$b$</td>
<td>$b$</td>
<td>$b \times b$ mat</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Stratified with N.S.</td>
<td>$b$</td>
<td>$\frac{b(b-1)}{s}$</td>
<td>$b$</td>
<td>$\frac{b}{s}$</td>
<td>$b \times \frac{b}{s}$ mat</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

- Computation cost estimation (using $b=256$, $k=20$, $t_f=10$, $t_g=100$, $t_i=1$, $s=2$)

  - IID sampling: 597k
  - Negative sampling: 546k
  - Stratified sampling (by item): 72k
  - Negative Sharing: 28k
  - Stratified sampling with negative sharing: 16k

   (all in time units)
Experimental Results

• Speedup up to 30 times with even performance improvement

• Datasets

Table 1: Data statistics for user, items and their interactions.

<table>
<thead>
<tr>
<th></th>
<th># of user</th>
<th># of item</th>
<th># of interaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Citeulike</td>
<td>5,551</td>
<td>16,980</td>
<td>204,986</td>
</tr>
<tr>
<td>News</td>
<td>10,000</td>
<td>58,579</td>
<td>515,503</td>
</tr>
</tbody>
</table>

Table 2: Data statistics for text content.

<table>
<thead>
<tr>
<th></th>
<th>voc. size</th>
<th>max</th>
<th>min</th>
<th>mean</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Citeulike (title)</td>
<td>4,777</td>
<td>15</td>
<td>2</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>Citeulike (title&amp;abs.)</td>
<td>23,011</td>
<td>300</td>
<td>22</td>
<td>194</td>
<td>186</td>
</tr>
<tr>
<td>News (title)</td>
<td>16,589</td>
<td>20</td>
<td>1</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>News (title&amp;sum.)</td>
<td>41,537</td>
<td>200</td>
<td>2</td>
<td>89</td>
<td>90</td>
</tr>
</tbody>
</table>
### Running Time

**Total speedup = speedup per iter * speedup of # iter**

#### Table 3: Comparisons of speedup for different sampling strategies against IID Sampling.

<table>
<thead>
<tr>
<th>Model</th>
<th>Sampling</th>
<th>CiteULike</th>
<th></th>
<th>News</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Per it.</td>
<td># of it.</td>
<td>Total</td>
<td>Per it.</td>
</tr>
<tr>
<td>CNN</td>
<td>Negative</td>
<td>1.02</td>
<td>1.00</td>
<td>1.02</td>
<td>1.03</td>
</tr>
<tr>
<td></td>
<td>Stratified</td>
<td>8.83</td>
<td>0.97</td>
<td>8.56</td>
<td>6.40</td>
</tr>
<tr>
<td></td>
<td>N.S.</td>
<td>8.42</td>
<td><strong>2.31</strong></td>
<td><strong>19.50</strong></td>
<td>6.54</td>
</tr>
<tr>
<td></td>
<td>Strat. w. N.S.</td>
<td><strong>15.53</strong></td>
<td>1.87</td>
<td><strong>29.12</strong></td>
<td><strong>11.49</strong></td>
</tr>
<tr>
<td>LSTM</td>
<td>Negative</td>
<td>0.99</td>
<td>0.96</td>
<td>0.95</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>Stratified</td>
<td>3.1</td>
<td>0.77</td>
<td>2.38</td>
<td>3.12</td>
</tr>
<tr>
<td></td>
<td>N.S.</td>
<td>2.87</td>
<td><strong>2.45</strong></td>
<td><strong>7.03</strong></td>
<td>2.78</td>
</tr>
<tr>
<td></td>
<td>Strat. w. N.S.</td>
<td><strong>3.4</strong></td>
<td><strong>2.22</strong></td>
<td><strong>7.57</strong></td>
<td><strong>3.13</strong></td>
</tr>
</tbody>
</table>

**Converge faster and fewer iterations are needed when more links are used!**
## Performance

### Table 4: Recall@50 for different sampling strategies under different models and losses.

<table>
<thead>
<tr>
<th>Model</th>
<th>Sampling</th>
<th>CiteULike</th>
<th>News</th>
</tr>
</thead>
<tbody>
<tr>
<td>IID</td>
<td></td>
<td>0.4746</td>
<td>0.4437</td>
</tr>
<tr>
<td></td>
<td>Negative</td>
<td>0.4725</td>
<td>0.4408</td>
</tr>
<tr>
<td>CNN</td>
<td>Stratified</td>
<td>0.4761</td>
<td>0.4394</td>
</tr>
<tr>
<td></td>
<td>Negative Sharing</td>
<td>0.4866</td>
<td>0.4423</td>
</tr>
<tr>
<td></td>
<td>Stratified with N.S.</td>
<td><strong>0.4890</strong></td>
<td><strong>0.4535</strong></td>
</tr>
<tr>
<td>LSTM</td>
<td>IID</td>
<td>0.4479</td>
<td>0.4718</td>
</tr>
<tr>
<td></td>
<td>Negative</td>
<td>0.4371</td>
<td>0.4668</td>
</tr>
<tr>
<td></td>
<td>Stratified</td>
<td>0.4344</td>
<td>0.4685</td>
</tr>
<tr>
<td></td>
<td>Negative Sharing</td>
<td>0.4629</td>
<td>0.4839</td>
</tr>
<tr>
<td></td>
<td>Stratified with N.S.</td>
<td><strong>0.4742</strong></td>
<td><strong>0.4877</strong></td>
</tr>
</tbody>
</table>
Convergence Curves

- Convergence to a better place using much less time

![Graph showing convergence curves for training and test phases.](image)
Summary

• Information network view of recommendation task
  • Capture context-rich environment

• Information network mining approaches can help recommendation tasks
  • Better performance and better interpretability

• Meta-path is powerful in capturing different intentions and similarities

• Sampling strategy becomes important when dealing with neural network-based collaborative filtering
References


• Y. Sun, Jiawei Han, Charu C. Aggarwal, and Nitesh V. Chawla. When Will It Happen? --- Relationship Prediction in Heterogeneous Information Networks. In WSDM, 2012.

• Y. Sun, R. Barber, M. Gupta, C. C. Aggarwal, and J. Han, Co-Author Relationship Prediction in Heterogeneous Bibliographic Networks. In ASONAM, 2011.


• X. Yu, X. Ren, Q. Gu, Y. Sun, and J. Han. Collaborative Filtering with Entity Similarity Regularization in Heterogeneous Information Networks. In IJCAI-HINA, 2013.

Outline

- Part I: Introduction and Preliminaries
- Part II: Recommendation in Heterogeneous Information Networks
- Part III: Recommendation in a Text-Rich Setting
- Part IV: Recommendation with Spatio-Temporal Information
- Part V: Research Frontiers and Summary
Break
PART III: RECOMMENDATION IN A TEXT-RICH SETTING

Xiang Ren
Department of Computer Science
University of Illinois at Urbana-Champaign / USC
xren7@illinois.edu

August 11, 2017
Outline

• Background

• Content-based Recommendation: An Overview

• Recommendation in Text-Rich Information Network

• Recommendation in Networks Constructed from Text

• Summary
Textual Information in Recommendation

- Rich text information are associated with users and items
  - User $\rightarrow$ textual user profile
  - Product / Movie $\rightarrow$ description, review
  - News article $\rightarrow$ textual content
  - Scientific paper $\rightarrow$ textual content
Use Case I: Recommending Related Articles

I wrote about the specific reasons we acquired Surphace here. Ultimately, it's about establishing a clear leader in the content discovery space, and providing a better service to readers and our partner publishers.


Exciting days!

YOU MIGHT ALSO LIKE

- Is Technology Killing Human Interaction?
- 11 Habits Of People Who Never Worry
- Who Says Laptops are Defunct...
- Architecture Students Join Forces
Use Case II: Movie Recommendation

More information about the movie.....

Recommendations

Challenge I

• **Unstructured textual information ➔ clean, structured representation?**

  • What are the semantic units?
    • Word, n-gram, phrases, entities, ...

  • Text is highly variable ➔ data sparsity
    • Domains, genres, languages

  • How to aggregate for objects (user, item, etc.)
    • Weighting methods?
Challenge II

• How to unify textual information with existing structured information?

Structured attribute information

Unstructured review text
How to Leverage Textual Information?

- Feature-based Approach
  - Content-based recommendation

- Network-based Approach
  - Recommendation in Text-Rich Information Network

- Text-to-Network Approach
  - Recommendation in Networks Constructed from Text
Outline

• Background

• Content-based Recommendation: An Overview

• Recommendation in Text-Rich Information Network

• Recommendation in Networks Constructed from Text

• Summary
Collaborative Filtering vs. Content-Based Recommendation

**COLLABORATIVE FILTERING**

- Read by both users
  - Similar users
  - Read by her, recommended to him!

**CONTENT-BASED FILTERING**

- Read by user
  - Similar articles
  - Recommended to user

---

https://www.themarketingtechnologist.co/building-a-recommendation-engine-for-geek-setting-up-the-prerequisites-13
Content-Based Recommendation: Basic Idea

User

Content used in the past

Rating

Feature values

User profile

Content with similar feature values is recommended.

Matching

Contents profile

Contents

Feature values

1 2 3

4 5 6
Advantages

• **User independence**
  - Item profile & user profile
  - No need for other users’ ratings (vs. CF)

• **Transparency**
  - Profile features → Why it is recommended?
  - CF: unknown users have similar tastes as yours

• **No “cold start”**
  - Effective on new items (if profile known)
Disadvantages

- **Restriction on content analysis**
  - Item profile is vague → low performance

- **“No surprise”**
  - Known features → no degree of “novelty”

- **New users**
  - Missing/incomplete user info → low performance
Major Components

**Content analyzer**

- Item $\rightarrow$ features
- Feature engineering, information extraction
Major Components

- **Content analyzer**
  - Item $\rightarrow$ features
  - Feature engineering, information extraction

- **Profiler learner**
  - User $\rightarrow$ feature profile
  - Data integration, user modeling
Major Components

- **Content analyzer**
  - Item $\rightarrow$ features
  - Feature engineering, information extraction

- **Profiler learner**
  - User $\rightarrow$ feature profile
  - Data integration, user modeling

- **Filtering component**
  - User $\rightarrow$ item recommendation
Content Analyzer: Item Representation

- **Items stored in a database table**

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Cuisine</th>
<th>Service</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1001</td>
<td>Mike’s Pizza</td>
<td>Italian</td>
<td>Counter</td>
<td>Low</td>
</tr>
<tr>
<td>1002</td>
<td>Chris’s Café</td>
<td>French</td>
<td>Table</td>
<td>Medium</td>
</tr>
<tr>
<td>1003</td>
<td>Jacques Bistro</td>
<td>French</td>
<td>Table</td>
<td>High</td>
</tr>
</tbody>
</table>

- **Structured data**
  - Small number of attributes
  - Each item is described by the same set of attributes
  - Known set of values that the attributes may have
Content Analyzer: Item Representation

- **Information about item could also be free text**
  - text description, customer review, news articles

- **Unstructured data**
  - No attribute names with well-defined values
  - Natural language complexity
    - Same word with different meanings
    - Different words with same meaning
**Item Representation: TF-IDF Weighting**

- Compute a weight for each term that represents the importance or relevance of that term
  - The term with highest weight occur more often in that document than in other documents
  - more central to the topic of the document
Item Representation: TF-IDF Weighting

- **Limitations**
  - This method does not capture the context in which a word is used
  - “This restaurant does not serve vegetarian dishes”

- **Information extraction**
  - turning text into machine-readable structures
User Profiles

- This profile consists of two main types of information
  - User’s interaction history.
    - items viewed by a user, items purchased by a user, search queries, etc.
  - A model of the user’s interests/preferences
    - \( s_{ij} = f(U_i, I_j) \) where \( U_i \) is user representation and \( I_j \) is item representation
    - \rightarrow How likely an user is interested in an item
User Interest Modeling

• “Manual” user interest modeling
  ▫ User customization
    • Provide “check box” interface that let the users construct their own profiles of interests

Limitations
• Efforts from user
• Cannot cope with changes in user interests
• ...
User Interest Modeling

- **Learning a user interest model**
  - **Learning a classifier**
    - Decision tree, Naïve Bayes, SVM, NeuralNets
  - **Training data: user-item interaction history**
    - Explicit ratings, implicit feedbacks
    - Feature space $\leftrightarrow$ features for user/item representations

<table>
<thead>
<tr>
<th>Cuisine</th>
<th>Service</th>
<th>Cost</th>
<th>Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Italian</td>
<td>Counter</td>
<td>Low</td>
<td>Negative</td>
</tr>
<tr>
<td>French</td>
<td>Table</td>
<td>Med</td>
<td>Positive</td>
</tr>
<tr>
<td>French</td>
<td>Counter</td>
<td>Low</td>
<td>Positive</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

```
Cuisine       Service Cost Rating
Italian Counter Low Negative
French Table Med Positive
French Counter Low Positive
```

---

Decision tree for user interest modeling:

- **Cuisine**
  - Italian
  - French
  - Mexican
- **Service**
  - Table
  - Counter
- **Cost**
  - High
  - Medium
  - Low
Content-Based Recommendation: Summary

• **Content-based Recommendation**
  - Basic Idea
  - Pros & cons
  - Major components

• **Item Representation**

• **User Profiles**
  ▫ Manual interest crafting
  • Learning A User Interest Model
Outline

• Background

• Content-based Recommendation: An Overview

• Recommendation in Text-Rich Information Network

• Recommendation in Networks Constructed from Text

• Summary
Representing Text: Two Approaches

• Feature-based Approach
  • Object (user, item, etc.) $\rightarrow$ feature representation
  • $\rightarrow$ Content-based recommendation models

“DBSCAN is a method for clustering in process of knowledge discovery.”
Representing Text: Two Approaches

• **Network-based Approach**
  - Information network as an unified data model
    - objects & text units $\rightarrow$ nodes
    - object-text unit relationships $\rightarrow$ edges
  - $\rightarrow$ Recommendation in Text-Rich Information Network
Network-Based Approach: Advantages

• **Unified representation**
  - Structured & unstructured information

• **Richer semantics**
  - Capture relationships between textual units

• **Collective inference**
  - Model objects jointly
Examples

Article / post Recommendation

Point-of-Interest Recommendation

Citation Recommendation
Citation Recommendation: Motivation

Personalized recommendation in social tagging systems using hierarchical clustering
A Shepitsten, J Gemmell, B Mobasher... - Proceedings of the 2008..., 2008 - dl.acm.org
... for profit or commercial advantage and that copies bear this notice and the full citation on the ... Other collaborative tagging applications focus on blogs, citations and wikis. ... if the personalized approach moves the resource further down the ranking in the recommendation set, the ...
Cited by 334 Related articles All 11 versions Cite Save

Context-aware citation recommendation
Q He, J Pei, D Kifer, P Mitra, L Giles - Proceedings of the 19th..., 2010 - dl.acm.org
... The bibliography candidates pro-vided by a global recommendation should collectively satisfy the citation information needs of all out-link ... Definition 3.3 (Local Recommendation). ... out-link local context c with respect to d, a local recom-mendation is a ranked list of citations in a ...
Cited by 94 Related articles All 16 versions Cite Save

Citation recommendation without author supervision
Q He, D Kifer, J Pei, P Mitra, CL Giles - ... on Web search and data mining, 2011 - dl.acm.org
... SETUP In this section, we introduce notation and terminology, and describe the citation recommendation problem. ... it to try to recognize locations in the query manuscript d where citations should exist. ... goal is to clus-ter this bipartite graph to obtain clusters of citation contexts and ...
Cited by 28 Related articles All 10 versions Cite Save
Citation Recommendation: Motivation

• Research papers need to cite relevant and important previous work
  • background, context and innovation

• Already large, rapidly growing body of scientific literature
  • automatic recommendations of high quality citations

• Traditional literature search systems
  • rich information needs → queries with a few keywords
Problem Statement

Paper titles, abstracts & bibliographic data

A new manuscript

SegPhrase [SIGMOD’15]

Author

Venue

Paper

“clustering”

“data mining”

“DB scan”

Author

Target venue

Phrases

ClusCite: Citation Recommendation by Information Network-Based Clustering [KDD’14]

Suggested papers to cite:

0.8

0.65

0.52
Prior Art: A Global Recommendation Model

Use Inference Network to integrate each hypothesis

- Content Match
- Publication Topical Prior
- Topic match

Given a paper abstract:
1. Word level match (language model)
2. Topic level match (KL-Divergence)
3. Topic importance

Prior Art: A Global Recommendation Model

Global Model: Limitations

**Global model:**
- all papers adopt same criterion and follow same behavioral pattern in citing other papers
  - e.g., equal importance between “content match” & “topic match” for every paper

- Context-based [He et al., WWW’10; Huang et al., CIKM’12]
- Topical similarity-based [Nallapati et al., KDD’08; Tang et al., PAKDD’09]
- Structural similarity-based [Liben-Nowell et al., CIKM’03; Strohman et al., SIGIR’07]
- Hybrid methods [Bethard et al., CIKM’10; Yu et al., SDM’12]
From Global Model to Paper-Specific Model

- **Global model:**
  - all papers adopt same criterion and follow same behavioral pattern in citing other papers

- **Paper citations → different interests groups**
  - Each group has its own behavioral pattern to identify references of interests
Each group follow distinct behavioral patterns and adopt different criterions in deciding relevance and authority of a candidate paper.
Each group follow distinct behavioral patterns and adopt different criterions in deciding relevance and authority of a candidate paper.
Distinctive Behavioral Pattern: Example

Each group follows distinct behavioral patterns and adopts different criteria in deciding relevance and authority of a candidate paper.
Distinctive Behavioral Pattern: Example

Each group follow distinct behavioral patterns and adopt different criterions in deciding relevance and authority of a candidate paper.
Distinctive Behavioral Pattern: Example

Each group follow distinct behavioral patterns and adopt different criterions in deciding relevance and authority of a candidate paper.
Each group **follow distinct behavioral patterns and adopt different criterions** in deciding relevance and authority of a candidate paper.
A unified graph representation for bibliographic dataset (papers and their attributes)

- Captures paper-paper relevance of different semantics
- Enables authority propagation between different types of objects
Citations tend to be *softly* clustered into different *interest groups*, based on the heterogeneous network structures

(Ren et al., KDD’15)
ClusCite: A Paper-specific Recommendation Model

Citations tend to be *softly* clustered into different *interest groups*, based on the heterogeneous network structures

| Learn distinct models on finding relevant papers and judging authority of papers |
| Derive group membership for query manuscript |

Paper-specific recommendation model: by integrating learned models of its related interest groups

Phrase I: Joint Learning (offline)  
Phrase II: Recommendation (online)

(Ren et al., KDD’15)
Proposed Model: Overview

How likely a query manuscript $q$ will cite a candidate paper $p$ (suppose $K$ interest groups):

$$s(q, p) = \sum_{k=1}^{K} \theta^{(k)}_{q} \cdot \left\{ r^{(k)}(q, p) + f_{p}^{(k)}(p) \right\}$$
Proposed Model: Overview

How likely a query manuscript $q$ will cite a candidate paper $p$ (suppose $K$ interest groups):

$$s(q, p) = \sum_{k=1}^{K} \frac{\theta_{q}^{(k)} \cdot \left[ r^{(k)}(q, p) + f^{(k)}_{p}(p) \right]}{\theta_{q}^{(k)}}$$

- **query’s group membership**
- **relative citation score (how likely $q$ will cite $p$) within each group**
Proposed Model: Overview

How likely a query manuscript $q$ will cite a candidate paper $p$ (suppose $K$ interest groups):

$$s(q, p) = \sum_{k=1}^{K} \theta^{(k)}_q \cdot \left\{ r^{(k)}(q, p) + f^{(k)}_P(p) \right\}$$

- **Query’s group membership**
- **Paper relative relevance** (query-candidate paper)
- **Paper relative authority** (candidate paper)
Proposed Model: Overview

How likely a query manuscript $q$ will cite a candidate paper $p$ (suppose $K$ interest groups):

$$s(q, p) = \sum_{k=1}^{K} \theta_q^{(k)} \cdot \left\{ r^{(k)}(q, p) + f_p^{(k)}(p) \right\}$$

- query’s group membership
- paper relative relevance (query-candidate paper)
- paper relative authority (candidate paper)

It is desirable to suggest papers that have *high* relevance and authority scores across *multiple* related interest groups of the query manuscript.
Proposed Model: Group Membership

- Learn each query’s group membership: **scalability & generalizability**
- Leverage the group memberships of related attribute objects to approximate query’s group membership

\[
s(q, p) = \sum_{k=1}^{K} \theta_q^{(k)} \cdot \left\{ r^{(k)}(q, p) + f_p^{(k)}(p) \right\}
\]

Different types of attribute objects (\(X = \text{authors/venues/terms}\))

Query’s related (linked) objects of type-\(X\)

Attribute object’s group membership (to learn)
Proposed Model: Paper Relevance

Network schema

\[
s(q, p) = \sum_{k=1}^{K} \theta_q^{(k)} \cdot \left\{ r^{(k)}(q, p) + f^{(k)}_p(p) \right\}
\]

\[
r^{(k)}(q, p) = \sum_{l=1}^{L} w_k^{(l)} \phi^{(l)}(q, p)
\]

Table 1: Meta paths with different semantics.

<table>
<thead>
<tr>
<th>Meta path</th>
<th>Semantic meaning of the relation</th>
</tr>
</thead>
<tbody>
<tr>
<td>(P - A - P)</td>
<td>(p_i) and (p_j) share same author(s)</td>
</tr>
<tr>
<td>(P - T - P)</td>
<td>(p_i) and (p_j) contain same term(s)</td>
</tr>
<tr>
<td>(P - V - P)</td>
<td>(p_i) and (p_j) are in the same venue</td>
</tr>
<tr>
<td>(P - T - P \rightarrow P)</td>
<td>(p_i) share term(s) with the paper(s) that cite (p_j)</td>
</tr>
<tr>
<td>(P - A - P \leftarrow P)</td>
<td>(p_i) share the same author(s) with the paper(s) cited by (p_j)</td>
</tr>
</tbody>
</table>
Proposed Model: Paper Relevance

Relevance features play different roles in different interest groups

\[ s(q, p) = \sum_{k=1}^{K} \theta_q^{(k)} \cdot \left\{ \tilde{r}^{(k)}(q, p) + f^{(k)}_\mathcal{P}(p) \right\} \]

\[ \tilde{r}^{(k)}(q, p) = \sum_{l=1}^{L} \underbrace{w_k^{(l)}}_{\text{weights on different meta path-based features}} \cdot \phi^{(l)}(q, p) \]

**Table 2: Learned weights on seven different meta paths for four mined interest groups (K = 40).**

<table>
<thead>
<tr>
<th>Meta path</th>
<th>Group 1</th>
<th>Group 2</th>
<th>Group 3</th>
<th>Group 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P - V - P )</td>
<td>0.0024</td>
<td>0.0113</td>
<td>0.0158</td>
<td>0.3076*</td>
</tr>
<tr>
<td>( P - A - P )</td>
<td>0.0054</td>
<td>0.0006</td>
<td>0.0192</td>
<td>0.1243</td>
</tr>
<tr>
<td>( P - A - P \rightarrow P )</td>
<td>0.6133**</td>
<td>0.2159*</td>
<td>0.2254</td>
<td>0.0213</td>
</tr>
<tr>
<td>( P - T - P )</td>
<td>0.1227</td>
<td>0.0947</td>
<td>0.1579</td>
<td>0.1095</td>
</tr>
<tr>
<td>( P - T - P \rightarrow P )</td>
<td>0.0442</td>
<td>0.5448**</td>
<td>0.3250*</td>
<td>0.0231</td>
</tr>
<tr>
<td>( P - T - P \leftarrow P )</td>
<td>0.1938*</td>
<td>0.0870</td>
<td>0.3578**</td>
<td>0.2409**</td>
</tr>
</tbody>
</table>
Proposed Model: Object Relative Authority

\[ s(q, p) = \sum_{k=1}^{K} \theta_q^{(k)} \cdot \left\{ r^{(k)}(q, p) + f_p^{(k)}(p) \right\} \]

**Paper relative authority:** A paper may have quite different visibility/authority among different groups, even it is overall highly cited
Proposed Model: Object Relative Authority

\[ s(q, p) = \sum_{k=1}^{K} \theta_q^{(k)} \cdot \left\{ r^{(k)}(q, p) + f_P^{(k)}(p) \right\} \]

**Paper relative authority:** A paper may have quite different visibility/authority among different groups, even it is overall highly cited.

### Table 6: Top-5 authority venues and authors from two example interest groups derived by ClusCite.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Group I (database and information system)</th>
<th>Author</th>
<th>Author Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>VLDB 0.0763</td>
<td>Hector Garcia-Molina</td>
<td>0.0202</td>
</tr>
<tr>
<td>2</td>
<td>SIGMOD 0.0653</td>
<td>Christos Faloutsos</td>
<td>0.0187</td>
</tr>
<tr>
<td>3</td>
<td>TKDE 0.0651</td>
<td>Elisa Bertino</td>
<td>0.0180</td>
</tr>
<tr>
<td>4</td>
<td>CIKM 0.0590</td>
<td>Dan Suciu</td>
<td>0.0179</td>
</tr>
<tr>
<td>5</td>
<td>SIGKDD 0.0488</td>
<td>H. V. Jagadish</td>
<td>0.0178</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rank</th>
<th>Group II (computer vision and multimedia)</th>
<th>Author</th>
<th>Author Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TPAMI 0.0733</td>
<td>Richard Szeliski</td>
<td>0.0139</td>
</tr>
<tr>
<td>2</td>
<td>ACM MM 0.0533</td>
<td>Jitendra Malik</td>
<td>0.0122</td>
</tr>
<tr>
<td>3</td>
<td>ICCV 0.0403</td>
<td>Luc Van Gool</td>
<td>0.0121</td>
</tr>
<tr>
<td>4</td>
<td>CVPR 0.0401</td>
<td>Andrew Blake</td>
<td>0.0117</td>
</tr>
<tr>
<td>5</td>
<td>ECCV 0.0393</td>
<td>Alex Pentland</td>
<td>0.0114</td>
</tr>
</tbody>
</table>

Relative authority propagation over the network:

- **WSDM**
- **KDD**
Model Learning: Joint Optimization

A joint optimization problem:

\[
\min_{P, W, F_P, F_A, F_V} \frac{1}{2} \mathcal{L} + \mathcal{R} + \frac{c_p}{2} \|P\|^2_F + \frac{c_w}{2} \|W\|^2_F \\
\text{s.t. } P \geq 0; \quad W \geq 0.
\]

\[
\mathcal{L} = \sum_{i,j=1}^{n} M_{ij} \left( Y_{ij} - \sum_{k=1}^{K} \sum_{l=1}^{L} \theta_{p_i}^{(k)} w_{k}^{(l)} S_{ji}^{(i)} - \sum_{k=1}^{K} \theta_{p_i}^{(k)} F_{P,kj} \right)^2
= \sum_{i=1}^{n} \left\| M_i \odot \left( Y_i - R_i P (WS^{(i)T} + F_P) \right) \right\|^2_2.
\]

\[
\mathcal{R} = \frac{\lambda_A}{2} \sum_{i=1}^{n} \sum_{j=1}^{A} R_{ij}^{(A)} \left\| \frac{F_{P,i}}{D_{ii}^{(P,A)}} - \frac{F_{A,i}}{D_{jj}^{(A,A)}} \right\|^2_2
+ \frac{\lambda_V}{2} \sum_{i=1}^{n} \sum_{j=1}^{|V|} R_{ij}^{(V)} \left\| \frac{F_{P,i}}{D_{ii}^{(P,V)}} - \frac{F_{V,i}}{D_{jj}^{(V,V)}} \right\|^2_2.
\]

Algorithm: alternating minimization (w.r.t. each variable)
Experimental Results

- Datasets
  - DBLP: 137k papers; ~2.3M relationships; Avg # citations/paper: 5.16
  - PubMed: 100k papers; ~3.6M relationships; Avg # citations/paper: 17.55
### Example output of relative authority ranking

<table>
<thead>
<tr>
<th>Rank</th>
<th>Venue</th>
<th>Author</th>
<th>Group I (database and information system)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>VLDB</td>
<td>Hector Garcia-Molina</td>
<td>0.0763</td>
</tr>
<tr>
<td>2</td>
<td>SIGMOD</td>
<td>Christos Faloutsos</td>
<td>0.0653</td>
</tr>
<tr>
<td>3</td>
<td>TKDE</td>
<td>Elisa Bertino</td>
<td>0.0651</td>
</tr>
<tr>
<td>4</td>
<td>CIKM</td>
<td>Dan Suciu</td>
<td>0.0590</td>
</tr>
<tr>
<td>5</td>
<td>SIGKDD</td>
<td>H. V. Jagadish</td>
<td>0.0488</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rank</th>
<th>Venue</th>
<th>Author</th>
<th>Group II (computer vision and multimedia)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TPAMI</td>
<td>Richard Szeliski</td>
<td>0.0733</td>
</tr>
<tr>
<td>2</td>
<td>ACM MM</td>
<td>Jitendra Malik</td>
<td>0.0533</td>
</tr>
<tr>
<td>3</td>
<td>ICCV</td>
<td>Luc Van Gool</td>
<td>0.0403</td>
</tr>
<tr>
<td>4</td>
<td>CVPR</td>
<td>Andrew Blake</td>
<td>0.0401</td>
</tr>
<tr>
<td>5</td>
<td>ECCV</td>
<td>Alex Pentland</td>
<td>0.0393</td>
</tr>
</tbody>
</table>
Experiment: Case Study II

- Case study on citation behavioral patterns

(a) Citations on venues

(b) Citations on authors

Each paper is assigned to the group with highest group membership score
Experiment: Comparing with State-of-the-Art Methods

Performance Comparisons

- 17.68% improvement in Recall@50; 9.57% in MRR, on DBLP

<table>
<thead>
<tr>
<th>Method</th>
<th>DBLP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P@10</td>
</tr>
<tr>
<td>BM25</td>
<td>0.1260</td>
</tr>
<tr>
<td>PopRank</td>
<td>0.0112</td>
</tr>
<tr>
<td>TopicSim</td>
<td>0.0328</td>
</tr>
<tr>
<td>Link-PLSA-LDA</td>
<td>0.1023</td>
</tr>
<tr>
<td>L2-LR</td>
<td>0.2274</td>
</tr>
<tr>
<td>RankSVM</td>
<td>0.2372</td>
</tr>
<tr>
<td>MixFea</td>
<td>0.2261</td>
</tr>
<tr>
<td>ClusCite-Rel</td>
<td>0.2402</td>
</tr>
<tr>
<td>ClusCite</td>
<td><strong>0.2429</strong></td>
</tr>
</tbody>
</table>

- BM25: content-based
- PopRank [WWW’05]: heterogeneous link-based authority
- TopicSim: topic-based similarity by LDA
- Link-PLSA-LDA [KDD’08]: topic and link relevance
- Meta-path based relevance:
  - L2-LR [SDM’12, WSDM’12]: logistics regression with L2 regularization
  - RankSVM [KDD’02]
- MixSim: relevance, topic distribution, PopRank scores, using RankSVM
Outline

• Background

• Content-based Recommendation: An Overview

• Recommendation in Text-Rich Information Network

• Recommendation in Networks Constructed from Text

• Summary
Text-Rich Information Network

• **Nodes:** phrases extracted from text
• **Edges:** relationship between phrase and document
  - How important is the phrase?
  - → TF-IDF weighting
  - → relationship strength (edge weight)

Entities Associated with Documents

- documents
- dbscan
- database
- clustering
- data mining
Network Construction from Text

• Can we construct the network from text?
  • First step: given nodes (phrases), can we learn the edge weights from data?

• Problem Statement
  • Joint learning of (1) recommendation model & (2) network edge weights for textual nodes
Example: Job Recommendation in LinkedIn

Given a LinkedIn member, we aim to find the jobs that he/she is most interested in.

Figure: Job recommendation panel on www.linkedin.com
Example: Job Recommendation in LinkedIn

What information is available for members and jobs?

<table>
<thead>
<tr>
<th>Content</th>
<th>Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recommender Systems for Talent Matching</td>
<td>Title</td>
</tr>
<tr>
<td>algorithms, python, machine learning, data mining, data analysis, linux, statistics</td>
<td>Skills</td>
</tr>
<tr>
<td>We use various data mining and information retrieval techniques to overcome the limitations of our sparse input data (member profiles and job descriptions)</td>
<td>Description</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Software Engineer Manager – Data Mining/Data Analysis/Machine Learning</td>
</tr>
<tr>
<td>Location</td>
<td>Bay Area, CA</td>
</tr>
<tr>
<td>Description</td>
<td>As a Software Engineer Data Manager, you will be responsible for leading a team of data scientists and relevance engineers that build and own recommendation algorithms, models, and systems.</td>
</tr>
<tr>
<td>Skills</td>
<td>java, C++, machine learning, data mining, information retrieval, big data, natural language processing, recommender systems</td>
</tr>
</tbody>
</table>
A Simple Solution: TF-IDF Weighting

- For each (member field \(s\), job field \(t\)), calculate the similarity score between two feature vectors.
- Aggregate the scores of all field pairs \((s, t)\).
Issue: Lower idf Terms Can be Predictive

- High idf term: government
- Low idf term: machine learning

Member (description)

Member
I have enrolled in a project which provides users with a visualization of government financial statistics using machine learning techniques ...

Recommended Jobs (description)

Job 1
We are a managed services provider and we support many projects with government agencies and non-profit organizations.

Job 2
You will apply machine learning algorithms to analyze large data sets ...
Issue: Lower idf Terms Can be Predictive

- High idf term: government
- Low idf term: machine learning

Limitations:
- The feature for each word is defined by heuristic, not necessarily reasonable
- Each field pair contributes equally
The Network View
Solution

• **Learn a better representation for words**
  - In particular, learn the optimal global term weights for each user text field and item text field
  - e.g., importance of “machine learning" in job skills

• **Learn the weights of multiple content matching features between user and item profiles (field pairs)**
  - e.g., user skills vs. job skills, user titles vs. job skills
A Two-layer Score Function Model

- **First layer:**
  - Map each original word feature into a scaled version
  - Calculate the cosine similar for each filed pair based on the weighted word feature

- **Second layer:**
  - Take the cosine similarity for each field pair as input, and take a weighted linear combination of these inputs
First Layer

- $V$: the size of the vocabulary
Second Layer

\[ s = \sum_{s,t} w^{(2)}_{st} \cdot \text{sim}_{s,t} + w^{(2)}_0 \]

Similarity between member field \( s \) and job field \( t \)
Objective: minimize the logit loss of all \((\text{member } i, \text{ job } j)\) pairs.

\[
L(W^{(1)}, W^{(2)}) = \sum_{i,j} \log(1 + e^{-y_{ij}s_{ij}})
\]
Experiments

**Data:**

- LinkedIn data
- 490K unique terms and 75 fields in total
- 3.1M (member, job) pairs

<table>
<thead>
<tr>
<th>Positive</th>
<th>Feedback Negative</th>
<th>Random Negative</th>
</tr>
</thead>
<tbody>
<tr>
<td>50%</td>
<td>25%</td>
<td>25%</td>
</tr>
</tbody>
</table>

- 90% as training, and 10% held out as testing
Case Study: Top Terms in Job Skills
Case Study: Top Terms in Member Skills

pharmacovigilance, Linux, forestry, Verilog, machine learning, design, logistics, talent, control.
Case Study: Most Important Member-Job Field Pairs

- Skill ID
- Skill Term
- Summary
- Past Position Summary
- Past Title

Member Fields

---

Job Fields

Skill ID
Skill Term
Description
Title
# AUC and AUPRC

<table>
<thead>
<tr>
<th>Method</th>
<th>AUC</th>
<th>AUPRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline (tf-idf as feature)</td>
<td>0.692</td>
<td>0.671</td>
</tr>
<tr>
<td>Multi-layer Logistic Regression Model</td>
<td>0.811 (+17.2%)</td>
<td>0.793 (+18.2%)</td>
</tr>
<tr>
<td>Multi-layer Logistic Regression Model (jobs only)</td>
<td>0.792 (+14.5%)</td>
<td>0.771 (+14.9%)</td>
</tr>
</tbody>
</table>

---

**ROC curve**

**Precision-Recall curve**
Outline

- Background

- Content-based Recommendation: An Overview

- Recommendation in Text-Rich Information Network

- Recommendation in Networks Constructed from Text

- Summary
Summary

- **Background**
  - Textual info in recommendation
  - Challenge: unstructured data to structures

- **Content-based Recommendation: An Overview**

- **Recommendation in Text-Rich Information Network**

- **Recommendation in Networks Constructed from Text**
Summary

• **Background**

• **Content-based Recommendation: An Overview**
  - Basic idea, pros & cons
  - Major components, item/user representations

• **Recommendation in Text-Rich Information Network**

• **Recommendation in Networks Constructed from Text**
Summary

• Background

• Content-based Recommendation: An Overview

• Recommendation in Text-Rich Information Network
  • Global recommendation model
  • Paper-specific recommendation model

• Recommendation in Networks Constructed from Text
Summary

• Background
• Content-based Recommendation: An Overview
• Recommendation in Text-Rich Information Network
• Recommendation in Networks Constructed from Text
  • A joint term-weight learning framework for recommendation
Outline

• Part I: Introduction and Preliminaries

• Part II: Recommendation in Heterogeneous Information Networks

• Part III: Recommendation in a Text-Rich Setting

• Part IV: Recommendation with Spatio-Temporal Information

• Part V: Research Frontiers and Summary
Break
PART IV: SPATIAL-TEMPORAL RECOMMENDATION

Hongzhi Yin
School of ITEE
University of Queensland, Australia
h.yin1@uq.edu.edu

August 11, 2017
Geo-Social Networks

- Geo-social network (GSN) is very popular
  - Location-based Social Networks - LBSNs (e.g., Foursquare, Instagram, Yelp, Facebook Places, Google Places)
  - Event-based Social Networks - EBSNs (e.g., Meetup, Plancast, Douban-Event)
  - Traditional Social Networks enhanced by locations (e.g., Sina Weibo, Twitter and Wechat)
Check-in in Geo-Social Networks

- Users can post their physical locations or geo-tag information via “check-in” and share their visiting experiences with their friends in the social networks.
- Check-in bridges the gap between Real World and Online Social Networks.
A check-in record consists of four elements: user, POI, time and check-in content.
Information Networks in Geo-social Networks

Gao et al. Data Analysis on Location-Based Social Networks
Spatial Item Recommendation

• **What to Recommend?**

• **Traditional Recommendation focuses on non-spatial items**
  - Virtual Items, i.e., items that can be digitalized, such as movie, music, news, webpage, games, apps
  - Products on E-Commerce websites

• **ST recommendation focuses on the recommendation of spatial items, i.e., items with geo-location attribute**
  - Point of Interests, such as restaurants, hotels, shops, stations
  - Events or Activities, such as party, concerts, culture salons, conferences and outdoor
Spatial Item Recommendation

- Spatial item recommendation aims to provide users valuable suggestions and assist them make right decision in their daily routines and trip planning, by sensing and mining
  - User Activities in the offline world
  - User Generated Contents in the online world

Geo-Social Networks can capture both.
Typical Recommendation Scenarios

- **Home-town Recommendation**
  - Make recommendations nearby users’ hometown or familiar regions
  - Most studies focus on.

- **Out-of-town Recommendation**
  - Make recommendation when users travel out of town or unfamiliar regions
  - More useful.

Yin. et al. LCARS: A location-content-aware recommendation system. (KDD’13)

Bao. et al. Location-based and preference-aware recommendation using sparse geo-social networking data. (SIGSPATIAL’12)
Outline

• Introduction

• Challenges of ST-Recommendation

• Effective Recommender Models
Data Sparsity and Travel Locality

- **Data Sparsity**
  - Millions of spatial items in the world
  - A user only check-ins a very small number of spatial items (less than 100), resulting in a very sparse user-item matrix.

- **Travel Locality**
  - Most of users’ check-in records are generated in their living regions (e.g., home cities), since users tend to travel a limited distance when visiting venues and attending events. User check-in records out-of-town are extremely sparse.
  - An investigation shows that the check-in records generated by users in their non-home cities are very few and only take up 0.47% of the check-in records they left in their home cities.
Example

Problem: When users from New York City are traveling in Los Angeles, how to make recommendations to them?

Yin. et al. LCARS: A location-content-aware recommendation system. (KDD’13)
Bao. et al. Location-based and preference-aware recommendation using sparse geo-social networking data. (SIGSPATIAL’12)
Travel Locality: When U3 travels to Los Angeles that is new to her since she has no activity history there, how can we recommend spatial items to her? In other words, how to link the users in one side to the items in the other side?

Both Graph-based methods and Collaborative Filtering methods would fail in this scenario.
Performance of CF-based Methods

1. CF performs well when the target locations are close to the home locations.
2. The precision degrades when the target locations are 100km away from their home locations.
3. The abrupt change at 100km can be explained by the fact that around 100 km is the typical human radius of “reach” as it takes about 1 to 2 hours to drive such distance.

Spatial Dynamics of User Interests

- Users tend to have different preferences when they travel in different regions, especially which have different urban compositions and cultures.
- For example, a user never goes gambling when she lives in Beijing, China, but when she travels in Macao or Las Vegas she is most likely to visit casinos.
- User preferences learned from her check-ins at one region (e.g., home city) are not necessarily applicable to other regions.
Spatial Dynamics of User Interests

• **Spatial Dynamics of User Interests**

  • We derive top four categories of POIs visited by a group of users in three different cities.

<table>
<thead>
<tr>
<th>City</th>
<th>Top POI Types</th>
<th>Percentage of Check-ins(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gold Coast (AU)</td>
<td>Beach</td>
<td>71.36%</td>
</tr>
<tr>
<td></td>
<td>Surf Spot</td>
<td>14.82%</td>
</tr>
<tr>
<td></td>
<td>Theme Park</td>
<td>9.60%</td>
</tr>
<tr>
<td></td>
<td>Scenic Lookout</td>
<td>3.36%</td>
</tr>
<tr>
<td>Las Vegas (US)</td>
<td>Casino</td>
<td>80.32%</td>
</tr>
<tr>
<td></td>
<td>Nightlife</td>
<td>10.61%</td>
</tr>
<tr>
<td></td>
<td>Outlet</td>
<td>5.82%</td>
</tr>
<tr>
<td></td>
<td>Hotel</td>
<td>3.23%</td>
</tr>
<tr>
<td>Istanbul (Turkey)</td>
<td>Mosque</td>
<td>68.32%</td>
</tr>
<tr>
<td></td>
<td>Museum</td>
<td>15.45%</td>
</tr>
<tr>
<td></td>
<td>Cafe</td>
<td>7.65%</td>
</tr>
<tr>
<td></td>
<td>Art Gallery</td>
<td>5.83%</td>
</tr>
</tbody>
</table>

Yin. et al. Adapting to user interest drift for POI recommendation. TKDE, 2016
Sequential Influence

- **Sequential Influence**
  - Human movement exhibits sequential patterns.
  - Besides personal interests, we also need to consider the spatial items the user has visited recently.

People usually go to cinemas or bars after restaurants since they would like to relax after dinner.

Temporal Dynamics

**Temporal Dynamics of User Preferences**

- Generally, users tend to have different needs and preferences at different times.
- A user is more likely to go to a restaurant rather than a bar for lunch at noon, and is more likely to go to a bar rather than a library at midnight.

User preference similarities between a given hour (6:00, 8:00, and 16:00) and other hours

Yuan et al. Time-aware Point-of-interest Recommendation. SIGIR-13
Temporal Dynamics

• A user’s preferences change continuously over time, but exhibits temporal cyclic patterns.
  • A user may regularly arrive at the office around 9:00 am, go to a restaurant for lunch at 12:00 pm, and watch movies at night around 8:00 pm

• There are multiple types of temporal cyclic patterns
  • Daily effect
  • Weekly effect
  • Weekday-Weekend pattern
  • Seasonal effect

• How to automatically choose the proper time granularity?
• How to implement a multi-granularity temporal model to automatically adapt to different datasets?

Hosseini. et al. Jointly Modelling Heterogeneous Temporal Properties in Location Recommendation (DASFAA-17)
Summary of Challenges

• Related with Spatial Factor
  • Data Sparsity
  • Travel Locality
  • Spatial Dynamics of User Interests;
    • Also called the drift of user interest across geographical regions

• Related with Temporal Factor
  • Sequential Influence
  • Multi-Granularity Temporal Cyclic Patterns

Out-of-town recommendation
A hard task!!!
References

References


- Saeid Hosseini, **Hongzhi Yin**, Meihui Zhang, Xiaofang Zhou and Shazia Sadiq. Jointly Modelling Heterogeneous Temporal Properties in Location Recommendation. (DASFAA'17)


Outline

• Introduction

• Challenges of ST-Recommendation

• Effective Recommender Models
  • To Address the Challenges with Spatial Factors
  • To Address the Challenges with Temporal Factors
### Data Sparsity and Travel Locality

#### Problem:
When users travel to an unfamiliar region, how to make recommendations to them?

---

As only the user-item interaction matrix is not enough to identify and transfer user interests, we leverage the content information of spatial items as medium.

The users in one side and the items in the other side can be linked together by the item contents.
Leverage the Wisdom of Crowds

• Leveraging the wisdom of crowds to deal with issue of user interest drift
  • By analyzing the word-of-mouth opinions from people who have visited \( l \) before, i.e., when people travel in city \( l \), what do most of them do? Which POIs have they visited? Which events attended? Exploiting the crowd’s behaviors to overcome the data sparsity of individual users in the unfamiliar regions.
Intuitive Ideas

Main idea #1: Identify user interest according to contents of their visited spatial items.

Main idea #2: Discover the crowd’s preferences in each region.

Main idea #3: Combine personal interest & region-aware crowd’s preferences

Implementation of Intuitive Ideas

- How to represent a user’s personal interests?
- How to represent the crowd’s preferences with respect to a specific region?
- How to combine the two factors in a principal way to make recommendations?
Location-Content-Aware LDA Model

• How to represent a user’s personal interests?
  • A multinomial distribution over a set of topics

• How to represent the crowd’s preferences with respect to a specific region?
  • A multinomial distribution over a set of topics

• How to combine the two factors in a principal way?
  • By introducing a “switch” variable to indicate which factor will be used to generate the user’s current check-in behavior

• Using topics to characterize both user interests and crowd preferences.

How to represent a topic

- Topic representation in topic models (LDA, PLSA): a multinomial distribution over a set of words

<table>
<thead>
<tr>
<th>Topic 1</th>
<th>Topic 2</th>
<th>Topic 3</th>
<th>Topic 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>retrieval</td>
<td>mining</td>
<td>neural</td>
<td>web</td>
</tr>
<tr>
<td>information</td>
<td>data</td>
<td>learning</td>
<td>services</td>
</tr>
<tr>
<td>document</td>
<td>discovery</td>
<td>networks</td>
<td>semantic</td>
</tr>
<tr>
<td>query</td>
<td>databases</td>
<td>deep</td>
<td>services</td>
</tr>
<tr>
<td>text</td>
<td>rules</td>
<td>analog</td>
<td>peer</td>
</tr>
<tr>
<td>search</td>
<td>association</td>
<td>vlsi</td>
<td>ontologies</td>
</tr>
<tr>
<td>evaluation</td>
<td>patterns</td>
<td>neurons</td>
<td>rdf</td>
</tr>
<tr>
<td>user</td>
<td>frequent</td>
<td>gaussian</td>
<td>management</td>
</tr>
<tr>
<td>relevance</td>
<td>streams</td>
<td>network</td>
<td>ontology</td>
</tr>
</tbody>
</table>
LCA-LDA Model

- **Topic:** A topic $z$ in LCA-LDA correspond to two distributions $\phi_z$ and $\phi'_z$. The former is a multinomial distribution over items (item ID) and the latter is a distribution over content words.
- Enabling clustering of both content-similar and co-visited spatial items into the same topics with high probability
- Integrating both CF information and content information
- **User Interests:** The intrinsic interests of user $u$ are represented by $\theta_u$, a multinomial distribution over topics.
- **Crowd Preferences:** The crowd preferences w.r.t a region $l$ are represented by $\theta_l$, a multinomial distribution over topics.
- **“Switch” Variable:** A switch variable $s$ is introduced to indicate which factor will be responsible for generating the check-in.
The Generative Process of LCA-LDA

Algorithm 1: Probabilistic generative process in LCA-LDA

for each topic $z$ do
  Draw $\phi_z \sim \text{Dirichlet}(.|\beta)$;
  Draw $\phi'_z \sim \text{Dirichlet}(.|\beta')$;
end

for each $D_u$ in $D$ do
  for each record $(u, v_{ui}, l_{ui}, c_{ui}) \in D_u$ do
    Toss a coin $s_{ui}$ according to $\text{bernoulli}(s_{ui}) \sim \text{beta}(\gamma, \gamma')$
    if $s_{ui} = 1$ then
      Draw $\theta_u \sim \text{Dirichlet}(.|\alpha)$;
      Draw a topic $z_{ui} \sim \text{multi}(\theta_u)$ according to the interest of user $u$;
    end
    if $s_{ui} = 0$ then
      Draw $\theta'_u \sim \text{Dirichlet}(.|\alpha')$;
      Draw a topic $z_{ui} \sim \text{multi}(\theta'_u)$ according to the local preference of $l_{ui}$;
    end
  end
  Draw an item $v_{ui} \sim \text{multi}(\phi_{z_{ui}})$ from $z_{ui}$-specific spatial item distribution;
  Draw a content word $c_{ui} \sim \text{multi}(\phi'_{z_{ui}})$ from $z_{ui}$-specific content word distribution;
end
Structure of LCA-LDA

\[ P(v|\theta_u, \theta_l^\text{crowd}) = \lambda_u P(v|\theta_u) + (1 - \lambda_u) P(v|\theta_l^\text{crowd}) \]
Online Recommendation

- The model parameters in LCA-LDA are estimated by Gibbs sampling.
- Given a query $q=(u,l)$, the ranking score of each item $v$ is computed as the inner product of the two vectors:

$$S(q, v) = \sum_z F(v, z)W(q, z)$$

$$W(q, z) = \hat{\lambda}_u \hat{\theta}_{u z} + (1 - \hat{\lambda}_u)\hat{\theta}'_{l_u z}$$

$$F(v, z) = \begin{cases}  
\hat{\phi}_{z v} \sum_{c_v \in C_v} \hat{\phi}'_{z c_v} & v \in V_{l_u} \\
0 & v \notin V_{l_u} 
\end{cases}$$
Example: \( q=(u, l) \)  
Parameters = \{ \( \lambda_u = 0.4 \), \( \theta_{uz}, \theta'_{lz}, F_{zv} \) \}

<table>
<thead>
<tr>
<th>( \theta_u )</th>
<th>Arts</th>
<th>0.25</th>
</tr>
</thead>
<tbody>
<tr>
<td>Food</td>
<td>0.2</td>
<td></td>
</tr>
<tr>
<td>Shop</td>
<td>0.35</td>
<td></td>
</tr>
<tr>
<td>Night Life</td>
<td>0.2</td>
<td></td>
</tr>
</tbody>
</table>

\( \lambda_u \)

\( \theta'_l \)

<table>
<thead>
<tr>
<th>Arts</th>
<th>0.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Food</td>
<td>0.5</td>
</tr>
<tr>
<td>Shop</td>
<td>0.3</td>
</tr>
<tr>
<td>Night Life</td>
<td>0.1</td>
</tr>
</tbody>
</table>

\( W_q \)

<table>
<thead>
<tr>
<th>Arts</th>
<th>0.16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Food</td>
<td>0.38</td>
</tr>
<tr>
<td>Shop</td>
<td>0.32</td>
</tr>
<tr>
<td>Night Life</td>
<td>0.14</td>
</tr>
</tbody>
</table>

\( W_{qz} \)

\( F_{zv} \)

\( S(q,v) = W_{qz} \times \)

\( F_{zv} = 0.076 \)

197
Limitations of LCA-LDA Model

- Inference complexity
- LCA-LDA considers multiple factors by introducing the additional “switch” variable $s$ to decide whether a topic is drawn from the user’s interests or the crowd’s preferences.
  - We need to sample both switch and topic for each check-in record
- From the perspective of mixture models, it needs to estimate a mixture weight $\lambda_u$ for each user by the switch variable $s$.
- It is not only computationally expensive to learn personalized mixture weights for individual users but also difficult to learn these mixture weights accurately given sparse datasets.
Limitations of LCA-LDA Model

- LCA-LDA ignores the roles of users.
  - Users with different roles tend to have different preferences regarding a region, such as local people vs. tourists.

- The location $l$ in LCA-LDA is fixed granularity, such as city. When the location granularity changes, LCA-LDA model needs to be retrained from scratch.
  - That is very time-consuming and infeasible.
Geo-Sparse Additive Generative Model

- **LCA-LDA**
  \[
  P(z|\theta^{user}_u, \theta^{\text{crowd}}_i) = \lambda_u P(z|\theta^{user}_u) + (1 - \lambda_u) P(z|\theta^{\text{crowd}}_i)
  \]

  When the data is sparse, infer the mixture weight by the switch variable for each user is both expensive and inaccurate.

- **Geo-SAGE**
  \[
  P(z|\theta^{user}_u, \theta^{\text{crowd}}_i) = \frac{\exp(\theta^{user}_{u,z} + \theta^{\text{crowd}}_{i,z})}{\sum_{z'} \exp(\theta^{user}_{u,z'} + \theta^{\text{crowd}}_{i,z'})}
  \]

  We can combine generative facets through simple addition in log space, avoiding the need for latent switching variables

Eisenstein et al. Sparse Additive Generative Models of Text (ICML 11)
Wang et al. Geo-SAGEA Geographical Sparse Additive Generative Model for Spatial Item Recommendation (KDD’15)
Both the user’s interests and the crowd’s preferences are represented by a probabilistic distributions and the mixture occurs in the distribution.
Geo-SAGE introduce a background model to capture the Common interests or topics among all users.

Both the user’s interests and the crowd’s preferences are represented by a vector with zero-mean variation.

The key difference between LCA-LDA and Geo-SAGE is that the mixture occurs in terms of natural parameters of the exponential family rather than distribution.
Role-Aware Crowd’s Preferences

• Generally, the crowds with different roles tend to have different preferences.

• In Geo-SAGE, we refine the crowd’s preferences
  • Native preferences: the common preferences of local people
  • Tourist preferences: the common preferences of tourists
Generative Process of Geo-SAGE

For each user activity record \((u, v, l_v, W_v, s)\)

1. Draw a topic index \(z\) according to \(u\)'s interests and the role-aware crowd’s preferences.

\[
P(z_{u,i}|s_{u,i}, l_{u,i}, \theta^0, \theta^\text{user}, \theta^\text{native}, \theta^\text{tourist})
\]

\[
= P(z_{u,i}|\theta^0 + \theta^\text{user} + (1 - s_{u,i}) \times \theta^\text{native}_i + s_{u,i} \times \theta^\text{tourist}_i)
\]

2. Draw content words according to \(z\)'s distributions over the words

\[
P(w_{u,i,n}|\phi^0, z_{u,i}, \phi^\text{topic}) = P(w_{u,i,n}|\phi^0 + \phi^\text{topic}_{z_{u,i}})
\]

3. Draw a spatial item according to \(z\)'s distributions over the spatial items

\[
P(v_{u,i}|\psi^0, z_{u,i}, \psi^\text{topic}) = P(v_{u,i}|\psi^0 + \psi^\text{topic}_{z_{u,i}})
\]
Leveraging Spatial Correlation for Data Sparsity

When there are few check-ins in a region, the inference of the role-aware crowd’s preferences may be inaccurate for this region.

As close regions have similar urban compositions and cultures, crowd’s preferences at these regions should be similar. Thus, we can “borrow” the check-in records from nearby regions to smooth crowd preferences at region r.

1. Spatial pyramid: partition the whole area into grids of varying sizes at different levels
2. Representing each region by a path from root to its corresponding cell.
3. Based on the path-based region representation, a hierarchically additive framework to represent crowd’s preferences

$$\theta_{i,\text{native}} = \sum_{h=1}^{H} \theta_{i,h,\text{native}}$$

$$\theta_{i,\text{tourist}} = \sum_{h=1}^{H} \theta_{i,h,\text{tourist}}$$

Advantages of hierarchically additive representations

• **Alleviating Data Sparsity.**
  - if there are few or no check-in records at region \( l \), we can still infer crowd preferences based on the check-in data generated at \( l \)’s ancestor regions.

• **Can be seamlessly integrated into Geo-SAGE model.**
  - Sharing the same additive feature

• **When the granularity of regions changes, we do not need to retrain the model.**
Automatic Adaption to the changing region size

- Current scale
- Current location in different scales

Level 1
Level 2
Level 3
Level 4

Zoom in
Zoom out
Data sets

• **Foursquare**  
  Publicly available  
  Contain 483,813 check-in records of 4163 users who live in the California, USA

• **Twitter**  
  Publicly available  
  Contain 1,434,668 check-ins of 114,058 users who live across whole USA

• **Distributions**
Comparative Approaches

- State-of-Art Spatial Items Recommendation
  - LCA-LDA
    - Latent factor model, Does not distinguish locals from tourists, without SAGE
  - CKNN
    - Local experts based method
  - UPS-CF
    - Friend-based collaborative filtering method
- Variant versions of Geo-SAGE
  - Geo-SAGE-S1
    - Without crowd’s preferences
  - Geo-SAGE-S2
    - Ignore the crowd’s roles
  - Geo-SAGE-S3
    - Without spatial pyramid

1. Geo-SAGE and LCA-LDA perform much better than CKNN: leverage the crowd’s preferences to address the challenges of user interest drift.

2. Geo-SAGE and LCA-LDA perform much better than UPS-CF: exploit the content information to identify and transfer user interests to overcome the changes of data sparsity and travel locality.

3. Geo-SAGE performs much better than LCA-LDA: apply the sparse additive model; role-aware crowd’s preferences.
Impact of Different Factors

1. Geo-SAGE consistently performs better than the three variant versions: indicate the benefits brought by each factor.
   - Geo-SAGE-S1: without crowd’s preferences
   - Geo-SAGE-S2: ignore users’ role
   - Geo-SAGE-S3: without spatial pyramid

2. Geo-SAGE-S2 and Geo-SAGE-S3 always perform better than Geo-SAGE-S1: show the advantage of integrating the crowd’s preferences.

3. The performance gap in home-town recommendation is smaller than out-of-town recommendation: performance improvement become less obvious when people travel in home town.
Short Summary

- **Challenges Related with Spatial Factor**
  - **Travel Locality**
    - Exploiting the content information of spatial items to identity and transfer users’ intrinsic interests
  - **Spatial Dynamics of User Interests**
    - Exploiting role-aware crowd’s preferences with respect to each region
  - **Data Sparsity**
    - Leveraging the spatial auto-correlation to borrow check-in data from other close regions

Out-of-town Recommendation
References

- Weiqing Wang, **Hongzhi Yin**, Ling Chen, Yizhou Sun and Xiaofang Zhou. ST-SAGE: A Spatial-Temporal Sparse Additive Generative Model for Spatial Item Recommendation (*ACM TIST’17*).
- Eisenstein et al. Sparse Additive Generative Models of Text (*ICML 11*).
Outline

• Introduction

• Challenges of ST-Recommendation

• Effective Recommender Models
  • To Address the Challenges with Spatial Factors
  • To Address the Challenges with Temporal Factors
Sequential Effect

• **Human movement exhibits sequential patterns** \[1, 2\]

• **Result from many factors**
  
  • **Temporal Effect, such as time in one day** \[3\]
    
    People tend to go to restaurants at dinner time and then relax in cinemas or bars at night.
  
  • **Geographical Influence, geographical proximity** \[4\]
    
    Tourists often sequentially visit London Eye, Big Ben and Downing Street.
  
  • **Other life-style related factors** \[5\]
    
    People usually check in at a Gym before a restaurant instead of the reverse way because it is not healthy to exercise right after a meal.


[4] Z. Yin, L. Gao, J. Han, J. Luo and T. S. Huang, ”Diversified trajectory pattern ranking in geo-tagged social media”, SIAM, 2011

Challenges-Modeling Sequential Influence

- The widely adopted Markov chain-based methods encounter the challenge of huge state prediction space

- Classical $n$th-order Markov chain
  - Predict the next possibly visiting spatial items based on all historical visited ones (Zhang et al., IEEE MDM, 2014).
  - Disadvantage: The prediction state space is $O(|V|^{n+1})$. When the number of items $|V|$ is slightly large, this method does not work.

- First-order Markov chain
  - Predict the next possibly visiting spatial item based on only the latest visited one (Chen et al., IEEE ICDE, 2011; Cheng et al., ACM MM, 2011; Cheng et al., IJCAI, 2013; Kurashima, ACM CIKM, 2010; Zheng et al., ACM TIST, 2012).
  - Disadvantage: Ignore the effect of other recent visited spatial items. Even in the first-order Markov chain model, the prediction state space is also very huge when there are millions of spatial items.
Challenges-Unifying Personalization and Sequential Effect

- **Unifying personalization and sequential effect**
  - Traditional recommender system
    - Focusing on personalization
    - Neglect the sequential effect
  - Existing sequential recommender system (i.e. Markov Chain)
    - Assume the same transition probabilities between items for all users
    - Ignore personalization
Main idea #1: Identify the personal interests of users based on topics (e.g., categories).

Main idea #3: Combine personal interests & sequential influence in the additive framework-SAGE

Main idea #2: Extracting sequential influence of all her recently visited spatial items in topic level instead of item level

Wang et al. SPORE: A sequential personalized spatial item recommender system. (ICDE’16)
We model personal interests and sequential influence based on the **latent variable topic-region** in SPOSE.

- A topic-region $z$ jointly corresponds to a semantic topic (i.e., a soft cluster of words describing spatial items, referring to categories) and a geographical region (i.e., a soft cluster of locations of spatial items).

- By introducing the topic-region, we decompose the spatial item prediction problem into two sub-problems:
  - predicting the topic-region $z$ of the user’s next activity based on her personal interests and her recently visited spatial items
  - then, predicting the next spatial items given the predicted topic-region $z$
Advantages by introducing latent Topic-Region

- Overcoming the data sparsity and low sampling rate
  - by focusing on the high-level topic-region rather than the fine granularity - spatial items
- Significantly reducing the prediction space
  - For each item $v$, we learn a distribution $\theta_v^{seq}$ over a set of topic-regions, and $\theta_{v,z}^{seq}$ represents the probability of visiting topic-region $z$ after visiting item $v$.
  - The number of topic-region is much smaller than the number of items, e.g., less than 100. Thus, the state space is reduced to $|V| \times K$ from $|V|^2$ compared with 1-order Markov chain model.
Additive Sequential Influence

- $P_u$ is the set of spatial items recently visited by $u$. Actually it is equivalent to a session in the online shopping scenario.
- How to combine the sequential influences from all items in $P_u$
  - Similar to the fusion of personal interests and sequential influence in SAGE model, we combine the sequential influence in the same way.
    \[
    P(z|\theta_{Pu}^{seq}) = \theta_{Pu,z}^{seq} = \frac{\exp(\sum_{v \in P_u} \theta_{v,z}^{seq})}{\sum_{z'} \exp(\sum_{v \in P_u} \theta_{v,z'}^{seq})}
    \]
  - Compared with classical n-order Markov chain model
    - We reduce the state parameter space from $|V|^{n+1}$ to $|V| \times K$
  - Compared with n-order additive Markov chain model (Lore) that works in a traditional mixture way
    - We reduce the state parameter space from $|V|^2$ to $|V| \times K$. Besides, we avoid to compute the mixture weights.

Lore: Exploiting sequential influence for location recommendations," in SIGSPATIAL, 2014
Generative Process of SPORE

- Draw a topic-region index $z_{u,i}$
  
  $z_{u,i} \sim P(z_{u,i} | P_{u,t_{u,i}}, \theta^0, \theta^{user}, \theta^{seq})$

- For each content word $w_{v_{u,i,n}}$ in $W_{v_{u,i}}$, draw
  
  $w_{v_{u,i,n}} \sim P(w_{v_{u,i,n}} | \phi^0, z_{u,i}, \phi^{topic})$

- Draw a spatial item $v_{u,i}$
  
  $v_{u,i} \sim P(v_{u,i} | \psi^0, z_{u,i}, \psi^{region})$

\[ P(z_{u,i} | P_{u,t_{u,i}}, \theta^0, \theta^{user}, \theta^{seq}) = P(z_{u,i} | \theta^0 + \theta^{user} + \theta^{seq}_{P_{u,t_{u,i}}}) \]

\[ P(v_{u,i} | \psi^0, z_{u,i}, \psi^{region}) = P(v_{u,i} | \psi^0 + \psi^{region}_{z_{u,i}}) \]

\[ P(w_{v_{u,i,n}} | \phi^0, z_{u,i}, \phi^{topic}) = P(w_{v_{u,i,n}} | \phi^0 + \phi^{topic}_{z_{u,i}}) \]
Limitation of SPORE

• SPORE ignores the effect of time on user mobility behaviors.

• Spatial item recommendation is a time-subtle recommendation task since at different time, users would prefer different successive POIs.
  • A user may go to a restaurant after leaving from office at noon, while he/she may be more likely to go to a gym when he/she leaves office at night.

Zhao et al. STELLAR: Spatial-Temporal Latent Ranking for Successive Point-of-Interest Recommendation. (AAAI’16)
Multi-Granularity Temporal Modeling

- There are multiple time granularities and various temporal cyclic patterns
  - Daily effect
  - Weekly effect
  - Weekday-Weekend pattern
  - Monthly effect
  - Seasonal effect
  - ...

- How to integrate various temporal cyclic patterns?
- How to implement a multi-granularity temporal model to automatically adapt to different datasets?
SAGE-Based Solution

- This method divides the time stamps according to the different granularities separately.
- In this way, each time stamp has multiple time ids with respect to the multiple granularities respectively.
- Then, combine the three temporal patterns in the additive SAGE framework.
  - add the effect of personal interests, multi-granularity temporal cyclic effect and sequential influence in the latent space

\[
P(z_{u,i} | P_{u,t_{u,i}}, \theta^\text{time}, \theta^\text{user}, \theta^\text{seq}) = P(z_{u,i} | \theta^\text{time}_{t_{u,i}} + \theta^\text{user}_u + \theta^\text{seq}_{P_{u,t_{u,i}}})
\]

\[
\theta^\text{time}_{t_{u,i}} = \theta^\text{year}_{t_{month}} + \theta^\text{week}_{t_{weekday}} + \theta^\text{day}_{t_{hour}}
\]

Experimental Results

TPM: Temporal-Sequential Model
TPM=SPORE + Temporal Influence

$TPM^1$ uses the hierarchical three-slice time indexing scheme (Zhao et al. AAAI16)
TPM uses our proposed additive time indexing scheme.
Short Summary

• **Challenges Related with Temporal Factor**
  - **Sequential Effect**
    - Low sampling in both time and space
    - Huge state prediction space
  - **Temporal Dynamics**
    - Multiple-granularity temporal cyclic patterns
    - How to integrate various temporal cyclic patterns to automatically adapt to different datasets
  - How to unify Personal Interests, Sequential Effect and Multi-granularity temporal cyclic patterns
References

Outline

• Part I: Introduction and Preliminaries

• Part II: Recommendation in Heterogeneous Information Networks

• Part III: Recommendation in a Text-Rich Setting

• Part IV: Recommendation with Spatio-Temporal Information

• Part V: Research Frontiers and Summary
Online Recommendation Efficiency

• **Real-time Response**
  - Given a mobile user, the naive approach to produce online top-k recommendations is to
    - first compute a score for each item
    - and then select k ones with highest scores.
    - However, when the number of available items becomes large, to produce a top-k ranked list using this brute-force method is very time-consuming and slow.
  - To support real-time recommendation in mobile scenario
    - Efficient smart online retrieval algorithms and effective indexing structures are required.
Recommended Reading

Online Recommendation Efficiency Issue:

Chapter 4: Fast Online Recommendation

To support real-time recommendation response, Smart retrieval algorithms + effective indexing structure

Threshold based algorithm (TA)
Attribute pruning-based algorithm (AP)
Metric-tree-based search algorithm (MP)
Locality-sensitive hashing (LSH)
Asymmetric Locality-sensitive hashing (ALSH)
Learning to hash techniques (L2H)
Cross-domain Recommendation

- Traditional recommender systems suggest items belonging to a single domain
  - movies in Netflix
  - songs in Last.fm
  - POIs in Foursquare

- In reality, users provide feedback for items of different types, and express their opinions on different social media and different providers
  - e.g., Facebook, Twitter, Amazon, Netflix

- Even items (or entities) from different domains and platforms are not independent or isolated
  - Sharing some attributes, semantics or hidden factors
  - Implicit links exist between these heterogeneous items
Can we leverage all the available personal data provided in distinct domains to generate better recommendations

- Linking Users across domains or platforms
- Multi-view user modeling

Can we exploit and leverage the common attributes, semantics and other hidden knowledge across distinct domains generate better recommendations?

- Very helpful to overcome the issue of the cold-start items
Cold-Start User in New Systems or Startup

- How to recommend items to new users?
- How to get user interest quickly?
- When new user comes, his feedback on what items can help us better understand his interest?
  - How to choose $k$ items to get most of the user’s interests?
    - Not very popular
    - Can represent a group of items
    - Users who like this item have different preference with users who dislike this item
    - The items that can reduce the entropy of the user’s interests to the maximum extent.
Summary

- Information Network Approach to model context-rich environment
- Recommendation Techniques in Heterogeneous Information Networks
- Recommendation Techniques in Text-Rich Setting
- Recommendation Techniques with Spatio-Temporal Information
Q & A

THANK YOU!