Collaborative Deep Learning for Recommender Systems

Hao Wang
Naiyan Wang
Dit-Yan Yeung
• Motivation

• Stacked Denoising Autoencoders

• Probabilistic Matrix Factorization

• Collaborative Deep Learning

• Experiments

• Summary
## Recommender Systems

### Motivation

### Stacked DAE

### PMF

### Collaborative DL

### Experiments

### Summary

**Rating matrix:**

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>✓</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>2</td>
<td>✓</td>
<td>?</td>
<td>?</td>
<td>✓</td>
<td>?</td>
</tr>
<tr>
<td>3</td>
<td>?</td>
<td>?</td>
<td>✓</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>4</td>
<td>?</td>
<td>✓</td>
<td>?</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>5</td>
<td>✓</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>

**Observed preferences:**

```
Matrix completion  Observed preferences: ✓
To predict: ?
```
Recommender Systems with Content

Content information:
Plots, directors, actors, etc.

Motivation
Stacked DAE  PMF  Collaborative DL  Experiments  Summary
Modeling the Content Information

- **Motivation**
- **Stacked DAE**
- **PMF**
- **Collaborative DL**
- **Experiments**
- **Summary**
Modeling the Content Information

1. Powerful features for content information
   - Deep learning

2. Feedback from rating information
   - Non-i.i.d.
     - Collaborative deep learning
Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction.

Bengio et al. 2015
Deep Learning

Stacked denoising autoencoders

Convolutional neural networks

Recurrent neural networks

Typically for i.i.d. data
Modeling the Content Information

1. Powerful features for content information
   - Deep learning

2. Feedback from rating information → Non-i.i.d.
   - Collaborative deep learning (CDL)
Contribution

• Collaborative deep learning:
  * deep learning for non-i.i.d. data
  * joint representation learning and collaborative filtering
Contribution

- Collaborative deep learning
- Complex target:
  * beyond targets like classification and regression
  * to complete a low-rank matrix
Contribution

- Collaborative deep learning
- Complex target
- First hierarchical Bayesian models for hybrid deep recommender system
Contribution

- Collaborative deep learning
- Complex target
- First hierarchical Bayesian models for hybrid deep recommender system
- Significantly advance the state of the art
• Motivation
• Stacked Denoising Autoencoders
• Probabilistic Matrix Factorization
• Collaborative Deep Learning
• Experiments
• Summary
Stacked Denoising Autoencoders (SDAE)

SDAE solves the following optimization problem:

$$\min_{\{W_l\}, \{b_l\}} \|X_c - X_L\|_F^2 + \lambda \sum_l \|W_l\|_F^2,$$

where $\lambda$ is a regularization parameter and $\| \cdot \|_F$ denotes the Frobenius norm.

Vincent et al. 2010
• Motivation
• Stacked Denoising Autoencoders
• Probabilistic Matrix Factorization
• Collaborative Deep Learning
• Experiments
• Summary
Probabilistic Matrix Factorization (PMF)

Graphical model:

\[ p(U | \sigma_U^2) = \prod_{i=1}^{N} \mathcal{N}(U_i | 0, \sigma_U^2 I) \]
\[ p(V | \sigma_V^2) = \prod_{j=1}^{M} \mathcal{N}(V_j | 0, \sigma_V^2 I) \]
\[ p(R | U, V, \sigma^2) = \prod_{i=1}^{N} \prod_{j=1}^{M} \left[ \mathcal{N}(R_{ij} | U_i^T V_j, \sigma^2) \right] I_{ij} \]

Objective function if using MAP:
\[ E = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{M} I_{ij} (R_{ij} - U_i^T V_j)^2 + \frac{\lambda_U}{2} \sum_{i=1}^{N} \| U_i \|_{Fro}^2 + \frac{\lambda_V}{2} \sum_{j=1}^{M} \| V_j \|_{Fro}^2 \]

Salakhutdinov et al. 2008

Notation:
- $v_j$ latent vector of item $j$
- $u_i$ latent vector of user $i$
- $r_{ij}$ rating of item $j$ from user $i$
• Motivation
• Stacked Denoising Autoencoders
• Probabilistic Matrix Factorization
• Collaborative Deep Learning
• Experiments
• Summary
Probabilistic SDAE

Graphical model:

Generative process:

\[ W_{l,*n} \sim \mathcal{N}(0, \lambda_{w}^{-1} I_{K_l}) \]
\[ b_{l} \sim \mathcal{N}(0, \lambda_{w}^{-1} I_{K_l}) \]
\[ X_{l,j*} \sim \mathcal{N}(\sigma(X_{l-1,j*} W_{l} + b_{l}), \lambda_{s}^{-1} I_{K_l}) \]
\[ X_{c,j*} \sim \mathcal{N}(X_{L,j*}, \lambda_{n}^{-1} I_{B}) \]
Collaborative Deep Learning

Motivation

- Two-way interaction
- More powerful representation
- Infer missing ratings from content
- Infer missing content from ratings

Notation:

- \( R \): rating of item j from user i
- \( x_0 \): corrupted input
- \( v \): latent vector of item j
- \( x_c \): clean input
- \( u \): latent vector of user i
- \( x_{L/2} \): content representation
- \( W^+ \): weights and biases

Graphical model:

Collaborative deep learning

SDAE
Collaborative Deep Learning

Information flows from ratings to content
Collaborative Deep Learning

Information flows from content to ratings
Collaborative Deep Learning

Reciprocal: representation and recommendation

Motivation | Stacked DAE | PMF | Collaborative DL | Experiments | Summary
Learning

maximizing the posterior probability is equivalent to maximizing the joint log-likelihood

\[ \mathcal{L} = -\frac{\lambda_u}{2} \sum_i \| u_i \|_2^2 - \frac{\lambda_w}{2} \sum_l (\| W_l \|_F^2 + \| b_l \|_2^2) \]

\[ -\frac{\lambda_v}{2} \sum_j \| v_j - X_{\frac{L}{2},j^*} \|_2^2 - \frac{\lambda_n}{2} \sum_j \| X_{L,j^*} - X_{c,j^*} \|_2^2 \]

\[ -\frac{\lambda_s}{2} \sum_l \sum_j \| \sigma(X_{l-1,j^*} W_l + b_l) - X_{l,j^*} \|_2^2 \]

\[ -\sum_{i,j} \frac{C_{ij}}{2} (R_{ij} - u_i^T v_j)^2. \]
Learning

Prior (regularization) for user latent vectors, weights, and biases

\[
\mathcal{L} = - \frac{\lambda_u}{2} \sum_i \| \mathbf{u}_i \|^2_2 - \frac{\lambda_w}{2} \sum_l (\| \mathbf{W}_l \|^2_F + \| \mathbf{b}_l \|^2_2) \\
- \frac{\lambda_v}{2} \sum_j \| \mathbf{v}_j - \mathbf{X}_{L \frac{L}{2}, j*} \|^2_2 - \frac{\lambda_n}{2} \sum_j \| \mathbf{X}_{L, j*} - \mathbf{X}_{C, j*} \|^2_2 \\
- \frac{\lambda_s}{2} \sum_l \sum_j \| \sigma(\mathbf{X}_{l-1, j*} \mathbf{W}_l + \mathbf{b}_l) - \mathbf{X}_{l, j*} \|^2_2 \\
- \sum_{i,j} \frac{C_{ij}}{2} (\mathbf{R}_{i,j} - \mathbf{u}_i^T \mathbf{v}_j)^2.
\]
Generating item latent vectors from content representation with Gaussian offset

\[ \mathcal{L} = -\frac{\lambda_u}{2} \sum_i \| u_i \|^2_2 - \frac{\lambda_w}{2} \sum_l (\| W_l \|^2_F + \| b_l \|^2_2) \]

\[ -\frac{\lambda_v}{2} \sum_j \| v_j - X^T_{\frac{L}{2}, j*} \|^2_2 - \frac{\lambda_n}{2} \sum_j \| X_{L, j*} - X_{c, j*} \|^2_2 \]

\[ -\frac{\lambda_s}{2} \sum_l \sum_j \| \sigma(X_{l-1, j*} W_l + b_l) - X_{l, j*} \|^2_2 \]

\[ - \sum_{i,j} \frac{C_{ij}}{2} (R_{i,j} - u_i^T v_j)^2. \]
Learning

‘Generating’ clean input from the output of probabilistic SDAE with Gaussian offset

\[
L = -\frac{\lambda_u}{2} \sum_i \|u_i\|^2_2 - \frac{\lambda_w}{2} \sum_l (\|W_l\|_F^2 + \|b_l\|_2^2) \\
- \frac{\lambda_v}{2} \sum_j \|v_j - X_{L/2,j*}^T\|^2_2 - \frac{\lambda_n}{2} \sum_j \|X_{L,j*} - X_{c,j*}\|_2^2 \\
- \frac{\lambda_s}{2} \sum_l \sum_j \|\sigma(X_{l-1,j*}W_l + b_l) - X_{l,j*}\|_2^2 \\
- \sum_{i,j} \frac{C_{ij}}{2}(R_{i,j} - u_i^T v_j)^2.
\]
Generating the input of Layer $l$ from the output of Layer $l-1$ with Gaussian offset

$$\mathcal{L} = -\frac{\lambda_u}{2} \sum_i \|u_i\|_2^2 - \frac{\lambda_w}{2} \sum_l (\|W_l\|_F^2 + \|b_l\|_2^2)$$

$$- \frac{\lambda_v}{2} \sum_j \|v_j - X_{L, j*}^T \|_2^2 - \frac{\lambda_n}{2} \sum_j \|X_{L, j*} - X_{c, j*}\|_2^2$$

$$- \frac{\lambda_s}{2} \sum_l \sum_j \|\sigma(X_{l-1, j*} W_l + b_l) - X_{l, j*}\|_2^2$$

$$- \sum_{i,j} \frac{C_{ij}}{2} (R_{i,j} - u_i^T v_j)^2.$$
Learning

measures the error of predicted ratings

\[ \mathcal{L} = -\frac{\lambda_u}{2} \sum_i \|u_i\|^2_2 - \frac{\lambda_w}{2} \sum_l (\|W_l\|^2_F + \|b_l\|^2_2) - \frac{\lambda_v}{2} \sum_j (\|v_j - X^T_{L/2,j*}\|^2_2 - \frac{\lambda_n}{2} \sum_j \|X_{L,j*} - X_{c,j*}\|^2_2) \]

\[ -\frac{\lambda_s}{2} \sum_l \sum_j \|\sigma(X_{l-1,j*}W_l + b_l) - X_{l,j*}\|^2_2 \]

\[ -\sum_{i,j} \frac{C_{ij}}{2} (R_{ij} - u_i^T v_j)^2. \]
Learning

If $\lambda_s$ goes to infinity, the likelihood becomes

$$
\mathcal{L} = - \frac{\lambda_u}{2} \sum_i \|u_i\|_2^2 - \frac{\lambda_w}{2} \sum_l (\|W_l\|_F^2 + \|b_l\|_2^2) \\
- \frac{\lambda_v}{2} \sum_j \|v_j - f_e(X_{0,j*}, W^+)^T\|_2^2 \\
- \frac{\lambda_n}{2} \sum_j \|f_r(X_{0,j*}, W^+) - X_{c,j*}\|_2^2 \\
- \sum_{i,j} \frac{C_{ij}}{2} (R_{ij} - u_i^T v_j)^2,
$$
Update Rules

For U and V, use block coordinate descent:

\[ u_i \leftarrow (VC_i V^T + \lambda_u I_K)^{-1} VC_i R_i \]

\[ v_j \leftarrow (UC_i U^T + \lambda_v I_K)^{-1} (UC_j R_j + \lambda_v f_e(X_{0,j*}, W^+)^T) \]

For W and b, use a modified version of backpropagation:

\[ \nabla_{w_l} \mathcal{L} = -\lambda_w W_l \]

\[ -\lambda_v \sum_j \nabla_{w_l} f_e(X_{0,j*}, W^+)^T (f_e(X_{0,j*}, W^+)^T - v_j) \]

\[ -\lambda_n \sum_j \nabla_{w_l} f_r(X_{0,j*}, W^+)(f_r(X_{0,j*}, W^+) - X_{c,j*}) \]

\[ \nabla_{b_l} \mathcal{L} = -\lambda_w b_l \]

\[ -\lambda_v \sum_j \nabla_{b_l} f_e(X_{0,j*}, W^+)^T (f_e(X_{0,j*}, W^+)^T - v_j) \]

\[ -\lambda_n \sum_j \nabla_{b_l} f_r(X_{0,j*}, W^+)(f_r(X_{0,j*}, W^+) - X_{c,j*}) \]
• Motivation
• Stacked Denoising Autoencoders
• Probabilistic Matrix Factorization
• Collaborative Deep Learning
• Experiments
• Summary
## Datasets

<table>
<thead>
<tr>
<th></th>
<th>citeulike-a</th>
<th>citeulike-t</th>
<th>Netflix</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>#users</strong></td>
<td>5551</td>
<td>7947</td>
<td>407261</td>
</tr>
<tr>
<td><strong>#items</strong></td>
<td>16980</td>
<td>25975</td>
<td>9228</td>
</tr>
<tr>
<td><strong>#ratings</strong></td>
<td>204987</td>
<td>134860</td>
<td>15348808</td>
</tr>
</tbody>
</table>

### Motivation

<table>
<thead>
<tr>
<th>Content information</th>
<th>Titles and abstracts</th>
<th>Titles and abstracts</th>
<th>Movie plots</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Wang et al. 2011</td>
<td>Wang et al. 2013</td>
<td></td>
</tr>
</tbody>
</table>

### Experiments

- **Stacked DAE**
- **PMF**
- **Collaborative DL**

### Summary
Evaluation Metrics

Recall:

\[
\text{recall}@M = \frac{\text{number of items that the user likes among the top } M}{\text{total number of items that the user likes}}
\]

Mean Average Precision (mAP):

\[
mAP = \frac{\sum_{q=1}^{Q} \text{AveP}(q)}{Q}
\]

\[
\text{AveP} = \frac{\sum_{k=1}^{n} (P(k) \times \text{rel}(k))}{\text{number of relevant items}}
\]

Higher recall and mAP indicate better recommendation performance
Comparing Methods

- **CMF**: Collective Matrix Factorization (Singh et al. 2008) is a model incorporating different sources of information by simultaneously factorizing multiple matrices.

- **SVDFeature**: SVDFeature (Chen et al. 2012) is a model for feature-based collaborative filtering.

- **DeepMusic**: DeepMusic (Oord et al. 2013) is a model for music recommendation.

- **CTR**: Collaborative Topic Regression (Wang et al. 2011) is a model performing topic modeling and collaborative filtering simultaneously.

Hybrid methods using BOW and ratings
Loosely coupled; interaction is not two-way
PMF+LDA
Recall@M

When the ratings are very sparse:

- citeulike-t, sparse setting
- Netflix, sparse setting

When the ratings are dense:

- citeulike-t, dense setting
- Netflix, dense setting
Mean Average Precision (mAP)

<table>
<thead>
<tr>
<th></th>
<th>citeulike-a</th>
<th>citeulike-t</th>
<th>Netflix</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDL</td>
<td>0.0514</td>
<td>0.0453</td>
<td>0.0312</td>
</tr>
<tr>
<td>CTR</td>
<td>0.0236</td>
<td>0.0175</td>
<td>0.0223</td>
</tr>
<tr>
<td>DeepMusic</td>
<td>0.0159</td>
<td>0.0118</td>
<td>0.0167</td>
</tr>
<tr>
<td>CMF</td>
<td>0.0164</td>
<td>0.0104</td>
<td>0.0158</td>
</tr>
<tr>
<td>SVDFeature</td>
<td>0.0152</td>
<td>0.0103</td>
<td>0.0187</td>
</tr>
</tbody>
</table>

Exactly the same as Oord et al. 2013, we set the cutoff point at 500 for each user.

A relative performance boost of about 50%
## Number of Layers

### Sparse Setting

<table>
<thead>
<tr>
<th>#layers</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>citeulike-a</td>
<td>27.89</td>
<td>31.06</td>
<td>30.70</td>
</tr>
<tr>
<td>citeulike-t</td>
<td>32.58</td>
<td>34.67</td>
<td>35.48</td>
</tr>
<tr>
<td>Netflix</td>
<td>29.20</td>
<td>30.50</td>
<td>31.01</td>
</tr>
</tbody>
</table>

### Dense Setting

<table>
<thead>
<tr>
<th>#layers</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>citeulike-a</td>
<td>58.35</td>
<td>59.43</td>
<td>59.31</td>
</tr>
<tr>
<td>citeulike-t</td>
<td>52.68</td>
<td>53.81</td>
<td>54.48</td>
</tr>
<tr>
<td>Netflix</td>
<td>69.26</td>
<td>70.40</td>
<td>70.42</td>
</tr>
</tbody>
</table>

The best performance is achieved when the number of layers is **2 or 3** (4 or 6 layers of generalized neural networks).
### Example User

<table>
<thead>
<tr>
<th></th>
<th># training samples</th>
<th>Movies</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTR</td>
<td>2</td>
<td>Swordfish, A Fish Called Wanda, Terminator 2, A Clockwork Orange, Sling Blade, Bridget Jones’s Diary, Raising Arizona, A Streetcar Named Desire, The Untouchables, The Full Monty</td>
</tr>
<tr>
<td>CDL</td>
<td>2</td>
<td>Snatch, The Big Lebowski, Pulp Fiction, Kill Bill, Raising Arizona, The Big Chill, Tootsie, Sense and Sensibility, Sling Blade, Swinger</td>
</tr>
</tbody>
</table>

**Top 10 recommended movies by CTR**
- Swordfish
- A Fish Called Wanda
- Terminator 2
- A Clockwork Orange
- Sling Blade
- Bridget Jones’s Diary
- Raising Arizona
- A Streetcar Named Desire
- The Untouchables
- The Full Monty

**Top 10 recommended movies by CDL**
- Snatch
- The Big Lebowski
- Pulp Fiction
- Kill Bill
- Raising Arizona
- The Big Chill
- Tootsie
- Sense and Sensibility
- Sling Blade
- Swinger

**Precison: 30% VS 20%**
### Example User

#### Action & Drama Movies

![Johnny English](image1.png)

**Johnny English**

**American Beauty**

![American Beauty](image2.png)

**American Beauty**

<table>
<thead>
<tr>
<th># training samples</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Top 10 recommended movies by CTR</strong></td>
<td></td>
</tr>
<tr>
<td>Pulp Fiction</td>
<td></td>
</tr>
<tr>
<td>A Clockwork Orange</td>
<td></td>
</tr>
<tr>
<td>Being John Malkovich</td>
<td></td>
</tr>
<tr>
<td><strong>Raising Arizona</strong></td>
<td></td>
</tr>
<tr>
<td>Sling Blade</td>
<td></td>
</tr>
<tr>
<td>Swordfish</td>
<td></td>
</tr>
<tr>
<td>A Fish Called Wanda</td>
<td></td>
</tr>
<tr>
<td>Saving Grace</td>
<td></td>
</tr>
<tr>
<td>The Graduate</td>
<td></td>
</tr>
<tr>
<td>Monster’s Ball</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th># training samples</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Top 10 recommended movies by CDL</strong></td>
<td></td>
</tr>
<tr>
<td>Pulp Fiction</td>
<td></td>
</tr>
<tr>
<td>Snatch</td>
<td></td>
</tr>
<tr>
<td><strong>The Usual Suspect</strong></td>
<td></td>
</tr>
<tr>
<td>Kill Bill</td>
<td></td>
</tr>
<tr>
<td>Memento</td>
<td></td>
</tr>
<tr>
<td>The Big Lebowski</td>
<td></td>
</tr>
<tr>
<td>One Flew Over the Cuckoo’s Nest</td>
<td></td>
</tr>
<tr>
<td>As Good as It Gets</td>
<td></td>
</tr>
<tr>
<td>Goodfellas</td>
<td></td>
</tr>
<tr>
<td>The Matrix</td>
<td></td>
</tr>
</tbody>
</table>

**Precision: 50% VS 20%**
## Example User

<table>
<thead>
<tr>
<th># training samples</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Top 10 recommended movies by CTR</strong></td>
<td></td>
</tr>
<tr>
<td>Best in Snow</td>
<td></td>
</tr>
<tr>
<td>Chocolat</td>
<td></td>
</tr>
<tr>
<td>Good Will Hunting</td>
<td></td>
</tr>
<tr>
<td>Monty Python and the Holy Grail</td>
<td></td>
</tr>
<tr>
<td>Being John Malkovich</td>
<td></td>
</tr>
<tr>
<td><em>Raising Arizona</em></td>
<td></td>
</tr>
<tr>
<td>The Graduate</td>
<td></td>
</tr>
<tr>
<td>Swordfish</td>
<td></td>
</tr>
<tr>
<td>Tootsie</td>
<td></td>
</tr>
<tr>
<td>Saving Private Ryan</td>
<td></td>
</tr>
<tr>
<td><strong>Top 10 recommended movies by CDL</strong></td>
<td></td>
</tr>
<tr>
<td>Good Will Hunting</td>
<td></td>
</tr>
<tr>
<td>Best in Show</td>
<td></td>
</tr>
<tr>
<td><em>The Big Lebowski</em></td>
<td></td>
</tr>
<tr>
<td>A Few Good Men</td>
<td></td>
</tr>
<tr>
<td>Monty Python and the Holy Grail</td>
<td></td>
</tr>
<tr>
<td><em>Pulp Fiction</em></td>
<td></td>
</tr>
<tr>
<td>The Matrix</td>
<td></td>
</tr>
<tr>
<td>Chocolat</td>
<td></td>
</tr>
<tr>
<td><em>The Usual Suspect</em></td>
<td></td>
</tr>
<tr>
<td>CaddyShack</td>
<td></td>
</tr>
</tbody>
</table>

**Precision: 90% VS 50%**
• Motivation
• Stacked Denoising Autoencoders
• Probabilistic Matrix Factorization
• Collaborative Deep Learning
• Experiments
• Summary
Non-i.i.d (collaborative) deep learning

With a complex target

First hierarchical Bayesian models for hybrid deep recommender system

Significantly advance the state of the art
Summary

- Word2vec, tf-idf
- Sampling-based, variational inference
- Tagging information, networks
Thank you!

Hao Wang
hwangaz@cse.ust.hk

More results, code, and datasets:
http://www.wanghao.in