Collaborative Deep Learning and Its Variants for Recommender Systems

Hao Wang

Joint work with Naiyan Wang, Xingjian Shi, and Dit-Yan Yeung

THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY
Recommender Systems

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>movie user</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<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>

Matrix completion

Observed preferences: ✅

To predict: ?
# Recommender Systems with Content

Content information:
Plots, directors, actors, etc.

<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><strong>user</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>movie</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<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>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>
Modeling the Content Information

Handcrafted features  
Automatically learn features 
Automatically learn features and adapt for ratings
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

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 deep hybrid recommender system
Related Work

• Not hybrid methods (ratings only)
  
  RBM (single layer, Salakhutdinov et al., 2007)
  
  I-RBM/U-RBM (Georgiev et al., 2013)

• Not using Bayesian modeling for joint learning

  DeepMusic (van den Oord et al., 2013)
  
  HLDBN (Wang et al., 2014)
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 ]
Probabilistic Matrix Factorization (PMF)

Graphical model:

Generative process:

\[
p(U | \sigma_U^2) = \prod_{i=1}^{N} \mathcal{N}(U_i | 0, \sigma_U^2 I) \quad \quad p(V | \sigma_V^2) = \prod_{j=1}^{M} \mathcal{N}(V_j | 0, \sigma_V^2 I) \]

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

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]
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 (CDL)

Graphical model:

Notation:
- $R$: rating of item $j$ from user $i$
- $V$: latent vector of item $j$
- $U$: latent vector of user $i$
- $x_0$: corrupted input
- $x_c$: clean input
- $w^i$: weights and biases
- $x_{L/2}$: content representation

Two-way interaction
- More powerful representation
- Infer missing ratings from content
- Infer missing content from ratings
A Principled Probabilistic Framework

Perception Component

Task-Specific Component

Perception Variables

Task Variables

Hinge Variables

[ Wang et al. TKDE 2016 ]
CDL with Two Components

Graphical model:

Collaborative deep learning

SDAE

Two-way interaction

- More powerful representation
- Infer missing ratings from content
- Infer missing content from ratings

Notation:
- $R_i$: rating of item $j$ from user $i$
- $x_0$: corrupted input
- $x_c$: clean input
- $v_j$: latent vector of item $j$
- $u_i$: latent vector of user $i$
- $W$: weights and biases
- $x_{L/2}$: content representation
Collaborative Deep Learning

Neural network representation for degenerated CDL
Collaborative Deep Learning

Information flows from ratings to content
Collaborative Deep Learning

Information flows from content to ratings
Collaborative Deep Learning

Representation learning <-> recommendation
Learning

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

\[ L = -\frac{\lambda_u}{2} \sum_i \| u_i \|^2_2 - \frac{\lambda_w}{2} \sum_l (\| W_l \|^2_F + \| b_l \|^2_2) 
- \frac{\lambda_v}{2} \sum_j \| v_j - X_{L, j^*}^T \|^2_2 - \frac{\lambda_n}{2} \sum_j \| X_{L, j^*} - X_{c, j^*} \|^2_2 
- \frac{\lambda_s}{2} \sum_l \sum_j \| \sigma(X_{l-1, j^*} W_l + b_l) - X_{l, j^*} \|^2_2 
- \sum_{i,j} \frac{C_{ij}}{2} (R_{i,j} - u_i^T v_j)^2. \]
Learning

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

\[
\mathcal{L} = \frac{-\lambda_u}{2} \sum_i \|u_i\|_2^2 - \frac{\lambda_w}{2} \sum_l (\|W_l\|_F^2 + \|b_l\|_2^2) \\
- \frac{\lambda_v}{2} \sum_j \|v_j - X_{L,j*}^{T/2}\|_2^2 - \frac{\lambda_n}{2} \sum_j \|X_{L,j*} - X_{c,j*}\|_2^2 \\
- \frac{\lambda_s}{2} \sum_l \sum_j \|\sigma(X_{l-1,j*} W_l + b_l) - X_{l,j*}\|_2^2 \\
- \sum_{i,j} \frac{C_{ij}}{2} (R_{i,j} - u_i^T v_j)^2.
\]
Learning

Generating item latent vectors from content representation with Gaussian offset

\[ \mathcal{L} = -\frac{\lambda_u}{2} \sum_i \| \mathbf{u}_i \|^2_2 - \frac{\lambda_w}{2} \sum_l (\| \mathbf{W}_l \|^2_F + \| \mathbf{b}_l \|^2_2) \]

\[ -\frac{\lambda_v}{2} \sum_j \| \mathbf{v}_j - \mathbf{X}^T_{L,j*} \|^2_2 \]

\[ -\frac{\lambda_n}{2} \sum_j \| \mathbf{X}_{L,j*} - \mathbf{X}_{c,j*} \|^2_2 \]

\[ -\frac{\lambda_s}{2} \sum_l \sum_j \| \sigma(\mathbf{X}_{l-1,j*} \mathbf{W}_l + \mathbf{b}_l) - \mathbf{X}_{l,j*} \|^2_2 \]

\[ -\sum_{i,j} \frac{C_{ij}}{2} (\mathbf{R}_{i,j} - \mathbf{u}_i^T \mathbf{v}_j)^2. \]
Learning

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

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

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

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

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

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

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

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

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

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

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

For U and V, use block coordinate descent:

\[ \mathbf{u}_i \leftarrow (\mathbf{V} \mathbf{C}_i \mathbf{V}^T + \lambda_u \mathbf{I}_K)^{-1} \mathbf{V} \mathbf{C}_i \mathbf{R}_i \]
\[ \mathbf{v}_j \leftarrow (\mathbf{U} \mathbf{C}_i \mathbf{U}^T + \lambda_v \mathbf{I}_K)^{-1} (\mathbf{U} \mathbf{C}_j \mathbf{R}_j + \lambda_v f_e(\mathbf{X}_{0,j*}, \mathbf{W}^+)^T) \]

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

\[ \nabla_{\mathbf{W}_l} \mathcal{L} = -\lambda_w \mathbf{W}_l \]
\[ -\lambda_v \sum_j \nabla_{\mathbf{W}_l} f_e(\mathbf{X}_{0,j*}, \mathbf{W}^+)^T (f_e(\mathbf{X}_{0,j*}, \mathbf{W}^+)^T - \mathbf{v}_j) \]
\[ -\lambda_n \sum_j \nabla_{\mathbf{W}_l} f_r(\mathbf{X}_{0,j*}, \mathbf{W}^+)(f_r(\mathbf{X}_{0,j*}, \mathbf{W}^+) - \mathbf{X}_{c,j*}) \]

\[ \nabla_{\mathbf{b}_l} \mathcal{L} = -\lambda_w \mathbf{b}_l \]
\[ -\lambda_v \sum_j \nabla_{\mathbf{b}_l} f_e(\mathbf{X}_{0,j*}, \mathbf{W}^+)^T (f_e(\mathbf{X}_{0,j*}, \mathbf{W}^+)^T - \mathbf{v}_j) \]
\[ -\lambda_n \sum_j \nabla_{\mathbf{b}_l} f_r(\mathbf{X}_{0,j*}, \mathbf{W}^+)(f_r(\mathbf{X}_{0,j*}, \mathbf{W}^+) - \mathbf{X}_{c,j*}) \]
# Datasets

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

## Content information

### Titles and abstracts

- [Wang et al. KDD 2011](#)
- [Wang et al. IJCAI 2013](#)

### Movie plots
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.
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>Method</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%
### Example User

#### Romance Movies

**Moonstruck**

- 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

**True Romance**

- Snatch
- **The Big Lebowski**
- Pulp Fiction
- Kill Bill
- **Raising Arizona**
- The Big Chill
- Tootsie
- Sense and Sensibility
- Sling Blade
- Swinger

<table>
<thead>
<tr>
<th># training samples</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top 10 recommended movies by CTR</td>
<td><strong>True Romance</strong></td>
</tr>
<tr>
<td># training samples</td>
<td>2</td>
</tr>
<tr>
<td>Top 10 recommended movies by CDL</td>
<td><strong>Moonstruck</strong></td>
</tr>
</tbody>
</table>

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

#### Action & Drama Movies

<table>
<thead>
<tr>
<th># training samples</th>
<th>Johnny English</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top 10 recommended movies by CTR</td>
<td></td>
</tr>
<tr>
<td>4</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>Raising Arizona</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>American Beauty</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top 10 recommended movies by CDL</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Pulp Fiction</td>
<td></td>
</tr>
<tr>
<td>Snatch</td>
<td></td>
</tr>
<tr>
<td>The Usual Suspect</td>
<td></td>
</tr>
<tr>
<td>Kill Bill</td>
<td></td>
</tr>
<tr>
<td>Momento</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: 20% VS 50%**
### Example User

<table>
<thead>
<tr>
<th># training samples</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top 10 recommended movies by CTR</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>Raising Arizona</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>
</tbody>
</table>

<table>
<thead>
<tr>
<th># training samples</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top 10 recommended movies by CDL</td>
<td></td>
</tr>
<tr>
<td>Good Will Hunting</td>
<td></td>
</tr>
<tr>
<td>Best in Show</td>
<td></td>
</tr>
<tr>
<td>The Big Lebowski</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>Pulp Fiction</td>
<td></td>
</tr>
<tr>
<td>The Matrix</td>
<td></td>
</tr>
<tr>
<td>Chocolat</td>
<td></td>
</tr>
<tr>
<td>The Usual Suspect</td>
<td></td>
</tr>
<tr>
<td>CaddyShack</td>
<td></td>
</tr>
</tbody>
</table>

**Precision: 50% VS 90%**
Summary: Collaborative Deep Learning

- 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
Marginalized CDL

\[ \mathcal{L} = -\frac{\lambda_u}{2} \sum_i \|u_i\|_2^2 - \frac{\lambda_v}{2} \sum_j \|v_j - f_e(X_{0,j^*}, W^+)^T\|_2^2 \]

CDL:

\[ -\frac{\lambda_n}{2} \sum_j \|f_r(X_{0,j^*}, W^+) - X_{c,j^*}\|_2^2 - \sum_{i,j} \frac{C_{ij}}{2} (R_{ij} - u_i^T v_j)^2 \]

Reconstruction error

Transformation to latent factors

Marginalized CDL:

\[ -\sum_j \|\tilde{X}_{0,j^*} W_1 - \tilde{X}_{c,j^*}\|_2^2 - \sum_{i,j} \frac{C_{ij}}{2} (R_{ij} - u_i^T v_j)^2 \]

Reconstruction error

Transformation to latent factors

\[ \mathcal{L} = -\frac{\lambda_u}{2} \sum_i \|u_i\|_2^2 - \frac{\lambda_v}{2} \sum_j \|v_j P_1 - X_{0,j^*} W_1\|_2^2 \]

[Li et al., CIKM 2015]
Collaborative Deep Ranking

Fig. 1. The graphic model of CDR. SDAE with \( L = 4 \) is presented inside the dashed rectangle. Note that \( W^+ \) denotes the set of weight matrices and bias vectors of all layers.

[Ying et al., PAKDD 2016]
Generative Process: Collaborative Deep Ranking

1. For each layer $l$ of the SDAE network,
   (a) For each column $q$, draw the weight matrix and bias vector $W_l^+$, draw $W_{l,*q}^+ \sim \mathcal{N}(0, \lambda_w^{-1}I_{K_l})$.
   (b) For each row $j$ of $X_l$, draw $X_{l,j*} \sim \mathcal{N}(\sigma(X_{l-1,j*}W_l + b_l), \lambda_s^{-1}I_{K_l})$

2. For each item $j$,
   (a) Draw a clean input $X_{c,j*} \sim \mathcal{N}(X_{L,j*}, \lambda_n^{-1}I_m)$
   (b) Draw a latent item offset vector $\epsilon_j \sim \mathcal{N}(0, \lambda_v^{-1}I_K)$ and then set the latent item vector to be:
      $$v_j = \epsilon_j + X_{L,\frac{L}{2},j*}^T$$

3. For each user $i$,
   (a) Draw user factor vector $u_i \sim \mathcal{N}(0, \lambda_u^{-1}I_K)$
   (b) For each pair-wise preference $(j, k) \in \mathcal{P}_i$, where $\mathcal{P}_i = \{(j, k) : r_{ij} - r_{ik} > 0\}$, draw the estimator,
      $$\delta_{ijk} \sim \mathcal{N}(u_i^T v_j - u_i^T v_k, c_{ijk}^{-1})$$
## CDL Variants

<table>
<thead>
<tr>
<th>CDL Variants</th>
<th>Venue</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deep CF</td>
<td>CIKM</td>
<td>2015</td>
</tr>
<tr>
<td>CD Ranking</td>
<td>PAKDD</td>
<td>2016</td>
</tr>
<tr>
<td>CF Networks</td>
<td>DLRS</td>
<td>2016</td>
</tr>
<tr>
<td>Collaborative KB Embedding</td>
<td>KDD</td>
<td>2016</td>
</tr>
<tr>
<td>AskGRU</td>
<td>RecSys</td>
<td>2016</td>
</tr>
<tr>
<td>ConvMF</td>
<td>RecSys</td>
<td>2016</td>
</tr>
<tr>
<td>Collaborative DAE</td>
<td>WSDM</td>
<td>2016</td>
</tr>
<tr>
<td>Collaborative Recurrent AE</td>
<td>NIPS</td>
<td>2016</td>
</tr>
<tr>
<td>DeepCoNN</td>
<td>WSDM</td>
<td>2017</td>
</tr>
<tr>
<td>Collaborative Metric Learning</td>
<td>WWW</td>
<td>2017</td>
</tr>
<tr>
<td>Additional SDAE</td>
<td>AAAI</td>
<td>2017</td>
</tr>
</tbody>
</table>

More details in [http://wanghao.in/CDL.htm](http://wanghao.in/CDL.htm)
Beyond Bag-of-Words: Documents as Sequences

**Motivation:**
- A more natural way, take in one word at a time, model documents as sequences
- Jointly model preferences and sequence generation under the BDL framework

“Collaborative recurrent autoencoder: recommend while learning to fill in the blanks” [Wang et al., NIPS 2016a]
Beyond Bag-of-Words: Documents as Sequences

Main Idea:
- Joint learning in the BDL framework
- Wildcard denoising for robust representation

“Collaborative recurrent autoencoder: recommend while learning to fill in the blanks”  [Wang et al., NIPS 2016a]
Wildcard Denoising

Sentence: This is a great idea. -> This is a great idea.

Direct Denoising:

Wrong transition

Wildcard Denoising:
Documents as Sequences

Main Idea:
- Joint learning in the BDL framework
- Wildcard denoising for robust representation
- Beta-Pooling for variable-length sequences

“Collaborative recurrent autoencoder: recommend while learning to fill in the blanks” [Wang et al., NIPS 2016a]
Is Variable-Length Weight Vector Possible?

<table>
<thead>
<tr>
<th>vector</th>
<th>length</th>
</tr>
</thead>
<tbody>
<tr>
<td>sequence</td>
<td>4</td>
</tr>
<tr>
<td>weight</td>
<td>8</td>
</tr>
</tbody>
</table>

- Length: 8

<table>
<thead>
<tr>
<th>vector</th>
<th>length</th>
</tr>
</thead>
<tbody>
<tr>
<td>sequence</td>
<td>3</td>
</tr>
<tr>
<td>weight</td>
<td>6</td>
</tr>
</tbody>
</table>

- Length: 6

<table>
<thead>
<tr>
<th>vector</th>
<th>length</th>
</tr>
</thead>
<tbody>
<tr>
<td>sequence</td>
<td>2</td>
</tr>
<tr>
<td>weight</td>
<td>4</td>
</tr>
</tbody>
</table>

- Length: 4

[Wang et al., NIPS 2016a]
Variable-Length Weight Vector with Beta Distributions

8 length-3 vectors

length-8 weight vector

one single vector

Use the area of the beta distribution to define the weights!

[Wang et al., NIPS 2016a]
Variable-Length Weight Vector with Beta Distributions

6 length-3 vectors

length-6 weight vector

one single vector

Use the area of the beta distribution to define the weights!

[Wang et al., NIPS 2016a]
Graphical Model: Collaborative Recurrent Autoencoder

- Joint learning in the BDL framework
- Wildcard denoising for robust representation
- Beta-Pooling for variable-length sequences

[Wang et al., NIPS 2016a]
Incorporating Relational Information

[ Wang et al. AAAI 2015 ]
[ Wang et al. AAAI 2017 ]
Probabilistic SDAE

Graphical model:

Generative process:

\[
\begin{align*}
W_{l,*n} & \sim \mathcal{N}(0, \lambda_w^{-1} I_{K_l}) \\
b_l & \sim \mathcal{N}(0, \lambda_w^{-1} I_{K_l}) \\
X_{l,j^*} & \sim \mathcal{N}(\sigma(X_{l-1,j^*} W_l + b_l), \lambda_s^{-1} I_{K_l}) \\
X_{c,j^*} & \sim \mathcal{N}(X_{L,j^*}, \lambda_n^{-1} I_B)
\end{align*}
\]

Generalized SDAE

Notation:
- \(x_0\): corrupted input
- \(x_c\): clean input
- \(W^+\): weights and biases
Relational SDAE: Graphical Model

Notation:
- $x_0$: corrupted input
- $X_c$: clean input
- $A$: adjacency matrix
Relational SDAE: Two Components

Perception Component

Task-Specific Component
Relational SDAE: Generative Process

1. Draw the relational latent matrix $S$ from a matrix variate normal distribution:

$$S \sim \mathcal{N}_{K,J}(0, I_K \otimes (\lambda_l \mathcal{L}_a)^{-1}).$$

2. For layer $l$ of the SDAE where $l = 1, 2, \ldots, \frac{L}{2} - 1$,
   1. For each column $n$ of the weight matrix $W_l$, draw $W_{l,n} \sim \mathcal{N}(0, \lambda_w^{-1} I_{K_l})$.
   2. Draw the bias vector $b_l \sim \mathcal{N}(0, \lambda_w^{-1} I_{K_l})$.
   3. For each row $j$ of $X_l$, draw

$$X_{l,j*} \sim \mathcal{N}(\sigma(X_{l-1,j*} W_l + b_l), \lambda_s^{-1} I_{K_l}).$$

3. For layer $\frac{L}{2}$ of the SDAE network, draw the representation vector for item $j$ from the product of two Gaussians (PoG):

$$X_{\frac{L}{2},j*} \sim \text{PoG}(\sigma(X_{\frac{L}{2}-1,j*} W_l + b_l), s^T_j, \lambda_s^{-1} I_{K}, \lambda_r^{-1} I_K).$$
Relational SDAE: Generative Process

1. For layer $l$ of the SDAE network where $l = \frac{L}{2} + 1, \frac{L}{2} + 2, \ldots, L$,
   - For each column $n$ of the weight matrix $W_l$, draw $W_{l,*n} \sim \mathcal{N}(0, \lambda_w^{-1}I_{K_l})$.
   - Draw the bias vector $b_l \sim \mathcal{N}(0, \lambda_w^{-1}I_{K_l})$.
   - For each row $j$ of $X_l$, draw $X_{l,j*} \sim \mathcal{N}(\sigma(X_{l-1,j*}W_l + b_l), \lambda_s^{-1}I_{K_l})$.

2. For each item $j$, draw a clean input $X_{c,j*} \sim \mathcal{N}(X_{L,j*}, \lambda_n^{-1}I_B)$. 
Multi-Relational SDAE: Graphical Model

Product of Q+1 Gaussians

Multiple networks:
citation networks
co-author networks

Notation:
- $x_0$: corrupted input
- $x_c$: clean input
- $A$: adjacency matrix
Relational SDAE: Objective Function

\[
\mathcal{L} = -\frac{\lambda_l}{2} \text{tr}(S \mathcal{L}_a S^T) - \frac{\lambda_r}{2} \sum_j \|(s_j^T - X_{L,j*})\|^2_2 - \frac{\lambda_w}{2} \sum_l (\|W_l\|^2_F + \|b_l\|^2_2)
\]

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

Similar to the generalized SDAE, taking \(\lambda_s\) to infinity, the joint log-likelihood becomes:

\[
\mathcal{L} = -\frac{\lambda_l}{2} \text{tr}(S \mathcal{L}_a S^T) - \frac{\lambda_r}{2} \sum_j \|(s_j^T - X_{L,j*})\|^2_2 - \frac{\lambda_w}{2} \sum_l (\|W_l\|^2_F + \|b_l\|^2_2)
\]

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

Network \(A \rightarrow\) Relational Matrix \(S\)

Relational Matrix \(S \rightarrow\) Middle-Layer Representations
Update Rules

For $S$:

$$S_{k*}(t + 1) \leftarrow S_{k*}(t) + \delta(t)r(t)$$

$$r(t) \leftarrow \lambda_r X_T^{T,*,k} - (\lambda_L \mathcal{L}_a + \lambda_r \mathbf{I}_J) S_{k*}(t)$$

$$\delta(t) \leftarrow \frac{r(t)^T r(t)}{r(t)^T (\lambda_L \mathcal{L}_a + \lambda_r \mathbf{I}_J) r(t)}.$$

For $X$, $W$, and $b$: Use Back Propagation.
From Representation to Tag Recommendation

Objective function:

$$\mathcal{L} = - \frac{\lambda_u}{2} \sum_i \|u_i\|^2_2 - \frac{\lambda_v}{2} \sum_j \|v_j - W^T_{T,j*}\|^2_2$$

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

where $\lambda_u$ and $\lambda_v$ are hyperparameters. $c_{ij}$ is set to 1 for the existing ratings and 0.01 for the missing entries.
Algorithm

1. **Learning representation:**
   
   repeat
   Update $S$ using the updating rules
   Update $X$, $W$, and $b$
   until convergence
   Get resulting representation $X_{\frac{L}{2},j*}$

2. **Learning $u_i$ and $v_j$:**
   
   Optimize the objective function $\mathcal{L}$

3. **Recommend tags to items according to the predicted $R_{ij}$:**
   
   $R_{ij} = u_i^T v_j$
   Rank $R_{1j}, R_{2j}, \ldots, R_{Ij}$
   Recommend tags with largest $R_{ij}$ to item $j$
## Datasets

Description of datasets

<table>
<thead>
<tr>
<th></th>
<th>citeulike-a</th>
<th>citeulike-t</th>
<th>movielens-plot</th>
</tr>
</thead>
<tbody>
<tr>
<td>#items</td>
<td>16980</td>
<td>25975</td>
<td>7261</td>
</tr>
<tr>
<td>#tags</td>
<td>7386</td>
<td>8311</td>
<td>2988</td>
</tr>
<tr>
<td>#tag-item paris</td>
<td>204987</td>
<td>134860</td>
<td>51301</td>
</tr>
<tr>
<td>#relations</td>
<td>44709</td>
<td>32665</td>
<td>543621</td>
</tr>
</tbody>
</table>
Sparse Setting, citeulike-a
Case Study 1: Tagging Scientific Articles

An example article with recommended tags

<table>
<thead>
<tr>
<th>Example Article</th>
<th>SDAE</th>
<th>True?</th>
<th>RSDAE</th>
<th>True?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title: Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews</td>
<td>1. instance</td>
<td>no</td>
<td>1. sentiment_analysis</td>
<td>no</td>
</tr>
<tr>
<td>Top topic 1: language, text, mining, representation, semantic, concepts, words, relations, processing, categories</td>
<td>2. consumer</td>
<td>yes</td>
<td>2. instance</td>
<td>no</td>
</tr>
<tr>
<td>3. sentiment_analysis</td>
<td>no</td>
<td>3. consumer</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>4. summary</td>
<td>no</td>
<td>4. summary</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>5. 31july09</td>
<td>no</td>
<td>5. sentiment</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>6. medline</td>
<td>no</td>
<td>6. product_review_mining</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>7. eit2</td>
<td>no</td>
<td>7. sentiment_classification</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>8. l2r</td>
<td>no</td>
<td>8. 31july09</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>9. exploration</td>
<td>no</td>
<td>9. opinion_mining</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>10. biomedical</td>
<td>no</td>
<td>10. product</td>
<td>yes</td>
<td></td>
</tr>
</tbody>
</table>

Precision: 10% VS 60%
Case Study 2: Tagging Movies (SDAE)

<table>
<thead>
<tr>
<th>Example Movie</th>
<th>Title: E.T. the Extra-Terrestrial</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Top topic 1: crew, must, on, earth, human, save, ship, rescue, by, find, scientist, planet</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Top 10 recommended tags</th>
<th>SDAE</th>
<th>True tag?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Saturn Award (Best Special Effects)</td>
<td>SDAE</td>
<td>yes</td>
</tr>
<tr>
<td>2. Want</td>
<td>SDAE</td>
<td>no</td>
</tr>
<tr>
<td>3. Saturn Award (Best Fantasy Film)</td>
<td>SDAE</td>
<td>no</td>
</tr>
<tr>
<td>4. Saturn Award (Best Writing)</td>
<td>SDAE</td>
<td>yes</td>
</tr>
<tr>
<td>5. Cool but freaky</td>
<td>SDAE</td>
<td>no</td>
</tr>
<tr>
<td>6. Saturn Award (Best Director)</td>
<td>SDAE</td>
<td>no</td>
</tr>
<tr>
<td>7. Oscar (Best Editing)</td>
<td>SDAE</td>
<td>no</td>
</tr>
<tr>
<td>8. almost favorite</td>
<td>SDAE</td>
<td>no</td>
</tr>
<tr>
<td>9. Steven Spielberg</td>
<td>SDAE</td>
<td>yes</td>
</tr>
<tr>
<td>10. sequel better than original</td>
<td>SDAE</td>
<td>no</td>
</tr>
</tbody>
</table>

Precision: 30% VS 60%
Case Study 2: Tagging Movies (RSDAE)

An example movie with recommended tags

<table>
<thead>
<tr>
<th>Example Movie</th>
<th>RSDAE</th>
<th>True tag?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top topic 1: crew, must, on, earth, human, save, ship, rescue, by, find, scientist, planet</td>
<td><strong>RSDAE</strong></td>
<td>yes</td>
</tr>
<tr>
<td><strong>Title: E.T. the Extra-Terrestrial</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. Steven Spielberg</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>2. Saturn Award (Best Special Effects)</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>3. Saturn Award (Best Writing)</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>4. Oscar (Best Editing)</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>5. Want</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>6. Liam Neeson</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>7. AFI 100 (Cheers)</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>8. Oscar (Best Sound)</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>9. Saturn Award (Best Director)</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td><strong>10. Oscar (Best Music - Original Score)</strong></td>
<td>yes</td>
<td></td>
</tr>
</tbody>
</table>

Does not appear in the tag lists of movies linked to ‘E.T. the Extra-Terrestrial’

Very difficult to discover this tag
Relational SDAE as Deep Relational Topic Models

BDL-Based Topic Models

Unified into a probabilistic relational model for relational deep learning

[ Wang et al. 2015 (AAAI) ]
(Recap) Relational SDAE: Two Components

**Perception Component**

- \( \lambda_w \)
- \( W^+ \)
- \( x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4 \rightarrow x_c \)
- \( \lambda_n \)

**Task-Specific Component**

- \( S \)
- \( A \)
- \( \lambda_l \)
- \( \lambda_r \)

**Diagram**

- Nodes: A, B, C, D, E, F, G, H, I, J
- Edges: A → B → C, D → E → F, J → G, I → H
Using Relational Information as Observations

Probabilistic SDAE

Modeling relation among nodes

[Wang et al. 2017 (AAAI) ]
Be ‘Bayesian’ in Collaborative Deep Learning
Be Bayesian in BDL

Motivation:
- Uncertainty estimation for reinforcement learning, active learning, etc.
- Robust for insufficient data and noise
- More accurate prediction

“Natural-Parameter Networks: A Class of Probabilistic Neural Networks”
Be Bayesian in BDL

What We Want:
- Solvable via back propagation
- Sampling-free during both training and testing
- Intuitive and easy to implement

“Natural-Parameter Networks: A Class of Probabilistic Neural Networks”
Weights/Neurons as Distributions

neural networks
weights/neurons as points

natural-parameter networks
weights/neurons as distributions
Take-home Messages

- Probabilistic graphical models for formulating both representation learning and inference/reasoning components
- Learnable representation serving as a bridge
- Tight, two-way interaction is crucial
Thanks!

Q&A