Towards Open-World Recommendation:
An Inductive Model-based Collaborative Filtering Approach

Qitian Wu, Hengrui Zhang, Xiaofeng Gao, Junchi Yan, Hongyuan Zha
Shanghai Jiao Tong University
The Chinese University of Hong Kong, Shenzhen
Background for Recommendation

- **Model-based** Collaborative Filtering ≈ Matric Factorization Model
- **Basic idea:**
  - user-item rating matrix
  - factorize
  - [user embeddings, item embeddings]
  - reconstruction matrix
- **CF models cannot handle new unseen users in open-world recommendation**

![Diagram](image-url)

Qitian Wu et al.  
Inductive Collaborative Filtering
Collaborative Filtering

- **Formulation of CF model for RecSys:**
  - A user-item interaction matrix $R = \{r_{ui}\}_{M \times N}$
  - Assume user latent factors $P = \{p_u\}_{M \times d}$
  - Assume item latent factors $Q = \{q_i\}_{N \times d}$
  - Consider an interaction model $\hat{r}_{ui} = f_\theta(p_u, q_i)$
  - Target objective $L(\hat{R}, R) = \sum_{(u,i)} L(\hat{r}_{ui}, r_{ui})$

- **Limitations: transductive learning**
  - Cannot handle new unseen users
    - Model retraining requires additional cost
    - Retraining may also lead to over-fitting
Challenges for Inductive Learning

- **Inductive learning** can be achieved via shared mapping

- **Expressiveness** would be sacrificed with inductive learning

**Pros**:
- Transductive learning: sufficient expressiveness
- Inductive learning: flexible for new users

**Cons**:
- Transductive learning: fail for new users
- Inductive learning: limited capacity/expressiveness
Related Works and Comparison

(a) General CF model
pros:
1. advanced capacity
2. fast training/inference
cons:
1. bad generalization
2. over-parametrization

(b) Item-based model
pros:
1. fewer parameters
2. enable inductive
cons:
1. limited capacity
2. user-item imbalance

(c) Local-graph-based inductive model
pros:
1. enable inductive
2. not require features
cons:
1. limited expressiveness
2. fail for implicit feedback

incremental learning
one-side learning
index-agnostic learning [ICLR’20]
Our Solutions: Inductive CF Model

- **Basic idea:**
  - leverage one group of users to express another
  - learn a latent graph over users
  - message passing from existing users to new ones

Key insight: user preferences share underlying proximity that induces latent graphs
Our Solutions: Inductive CF Model (Cont.)

Partition users into two groups: $|U_k| = M_k$ $|U_q| = M_q$

- **Key users:** transductive learning (traditional model)
  
  **model:** $P_k = \{p_u\}_{M_k \times d}$ $Q = \{q_i\}_{N \times d}$ $\hat{r}_{ui} = f_\theta(p_u, q_i)$

  **learning:** $\min_{P_k, Q, \theta} D_S(\hat{R}_k, R_k)$ where $R_k = \{r_{ui}\}_{M_k \times N}$

- **Query users:** inductive learning (new model)

  **model:** $\hat{P}_u' = c_w' P_k$ $c_w' u = \frac{e^T [W_q d_w' \oplus W_k p_u]}{\sum_{u_0 \in U_k} e^T [W_q d_w' \oplus W_k p_{u_0}]}$ where $d_w' = \sum_{i \in I_u} q_i$

  **learning:** $\min_{w, \theta} D_S(\hat{R}_q, R_q)$ where $R_q = \{r_{ui}\}_{M_q \times N}$ $\hat{r}_{ui} = f_\theta(\hat{P}_u, q_i)$

  **objective:** $\min_{w, \theta} D_S(\hat{R}_q, R_q) + \lambda L_C(P_k, \hat{P}_k)$ $L_C(P_k, \hat{P}_k) = \frac{1}{M_q} \sum_{u \in U_k} \log \frac{\exp(p_u^T \hat{p}_u)}{\sum_{u' \in U_q} \exp(p_{u'}^T \hat{p}_{u'})}$

  **regularization:** consistency between two estimated embeddings for one user
Our Solutions: Inductive CF Model (ont.)

- Learning procedures: pretraining + adaption
- Consider two scenarios in open-world recommendation:
  - few-shot users: pretrained model fine-tunes on new users' data
  - zero-shot users: pretrained model directly operate on new users

(a) Inductive learning for interpolation
(b) Inductive learning for extrapolation
Theoretical Analysis

- The model possesses the same representation capacity compared to matrix factorization
  - The only mild condition is that key users' latent factors span the latent space
- The generalization ability on new users depends on number of key users and training instances of new users

**Theorem 1.** Assume Eq. (3) can achieve $D_{S_q}(\hat{R}_q, R_q) < \epsilon$ and the optimal $P_k$ given by Eq. (1) satisfies column-full-rank, then there exists at least one solution for $C$ in Eq. (2) such that $D_{S_q}(\hat{R}_q, R_q) < \epsilon$.

\[
\min_{P_k, Q, \theta} D_{S_k}(\hat{R}_k, R_k), \quad (1)
\]

\[
\min_{C, Q} D_{S_q}(\hat{R}_q, R_q), \quad (2)
\]

\[
\min_{P_q, Q} D_{S_q}(\hat{R}_q, R_q), \quad (3)
\]

**Theorem 2.** Assume 1) $D$ is $L$-Lipschitz, 2) for $\forall \hat{r}_{u'i} \in \hat{R}_q$ we have $|\hat{r}_{u'i}| \leq B$, and 3) the $L1$-norm of $c_{u'}$ is bounded by $H$. Then with probability at least $1 - \delta$ over the random choice of $S_q \in ([M_q] \times [N])^{T_q}$, it holds that for any $\hat{R}_q$, the gap between $D(\hat{R}_q, R_q)$ and $D_{S_q}(\hat{R}_q, R_q)$ will be bounded by

\[
O \left( 2LHB \sqrt{\frac{2M_q \ln M_k}{T_q}} + \sqrt{\ln(1/\delta)} \right). \quad (8)
\]
Experiments Setup

- **Dataset information:**

<table>
<thead>
<tr>
<th>Dataset</th>
<th># Users</th>
<th>#Items</th>
<th># Ratings</th>
<th>Density</th>
<th># Key/Query Users</th>
<th># Training/Test Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Douban</td>
<td>3,000</td>
<td>3,000</td>
<td>0.13M</td>
<td>0.0152</td>
<td>2,131/869</td>
<td>80,000/20,000</td>
</tr>
<tr>
<td>Movielens-100K</td>
<td>943</td>
<td>1,682</td>
<td>0.10M</td>
<td>0.0630</td>
<td>123,202/13,689</td>
<td></td>
</tr>
<tr>
<td>Movielens-1M</td>
<td>6,040</td>
<td>3,706</td>
<td>1.0M</td>
<td>0.0447</td>
<td>5,114/926</td>
<td>900,199/100,021</td>
</tr>
<tr>
<td>Amazon-Books</td>
<td>52,643</td>
<td>91,599</td>
<td>2.1M</td>
<td>0.0012</td>
<td>49,058/3,585</td>
<td>2,405,036/526,430</td>
</tr>
<tr>
<td>Amazon-Beauty</td>
<td>2,944</td>
<td>57,289</td>
<td>0.08M</td>
<td>0.0004</td>
<td>780/2,164</td>
<td>53,464/29,440</td>
</tr>
</tbody>
</table>

- **Evaluation Protocol:**
  - **Explicit dataset:** random split, RMSE & NDCG metric
  - **Implicit dataset:** leave-last-out, AUC & NDCG metric, negative sampling

- **Comparison:** CF models, inductive models, feature-based models
Experiments Setup

- **Implementation:**
  - **IDCF-NN:** feedforward neural network as predictor
    \[
    f_\theta(p_u, q_i) = \frac{(p_u^T q_i + nn([p_u \parallel q_i] || p_u \odot q_i))}{2} + b_u + b_i
    \]
  - **IDCF-GC:** graph convolution network as predictor
    \[
    m_{u,m} = ReLU\left(\frac{1}{|N_{u,m}|} \sum_{i \in N_{u,m}} W_{q,m} q_i\right)
    \]
    \[
    n_{i,m} = ReLU\left(\frac{1}{|N_{i,m}|} \sum_{u \in N_{i,m}} W_{p,m} p_u\right)
    \]
    \[
    f(p_u, q_i, \{p_u\}_{u \in N_i}, \{q_i\}_{i \in N_u}) = nn'([p_u \odot q_i || p_u \odot m_u || n_i \odot q_i || n_i \odot m_u]) + b_u + b_i
    \]
Experiments

- **Comparison results for explicit feedback:**
  - For few-shot query users, very **competitive** results as inductive models and very **close** test performance to transductive models
  - For zero-shot new users, significantly outperform **SOTA** inductive models

<table>
<thead>
<tr>
<th>Method</th>
<th>Inductive</th>
<th>Feature</th>
<th>Douban RMSE</th>
<th>Douban NDCG</th>
<th>ML-100K RMSE</th>
<th>ML-100K NDCG</th>
<th>ML-1M RMSE</th>
<th>ML-1M NDCG</th>
</tr>
</thead>
<tbody>
<tr>
<td>PMF</td>
<td>No</td>
<td>No</td>
<td>0.737</td>
<td>0.718</td>
<td>0.939</td>
<td>0.954</td>
<td>0.932</td>
<td>1.003</td>
</tr>
<tr>
<td>NNMF</td>
<td>No</td>
<td>No</td>
<td>0.729</td>
<td><strong>0.705</strong></td>
<td>0.939</td>
<td>0.952</td>
<td>0.925</td>
<td>0.987</td>
</tr>
<tr>
<td>GCMC</td>
<td>No</td>
<td>No</td>
<td>0.731</td>
<td>0.706</td>
<td>0.938</td>
<td><strong>0.956</strong></td>
<td>0.911</td>
<td>0.989</td>
</tr>
<tr>
<td>NIMC</td>
<td>Yes</td>
<td>Yes</td>
<td>0.732</td>
<td>0.745</td>
<td>0.928</td>
<td>0.931</td>
<td>1.015</td>
<td>1.065</td>
</tr>
<tr>
<td>BOMIC</td>
<td>Yes</td>
<td>Yes</td>
<td>0.735</td>
<td>0.747</td>
<td>0.923</td>
<td>0.925</td>
<td>0.931</td>
<td>1.001</td>
</tr>
<tr>
<td>F-EAE</td>
<td>Yes</td>
<td>No</td>
<td>0.738</td>
<td>-</td>
<td>-</td>
<td>0.920</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>IGMC</td>
<td>Yes</td>
<td>Yes</td>
<td>0.721</td>
<td><strong>0.728</strong></td>
<td>-</td>
<td>0.905</td>
<td><strong>0.997</strong></td>
<td>-</td>
</tr>
</tbody>
</table>

**IDCF-NN (ours)** Yes No 0.738 0.712 0.939 0.956 0.931 0.996 0.896 0.880 0.844 0.952 0.922 **0.940**

**IDCF-GC (ours)** Yes No 0.733 0.712 **0.940** 0.956 **0.905** 0.981 0.901 0.884 0.839 0.944 **0.924** 0.940

Lower RMSE and higher NDCG are better
Experiments

- Comparison results for implicit feedback:
  - For few-shot query users, achieve SOTA results
  - For zero-shot new users, significantly improvement

<table>
<thead>
<tr>
<th>Method</th>
<th>Amazon-Books</th>
<th>Amazon-Beauty</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AUC</td>
<td>Query</td>
</tr>
<tr>
<td>PMF</td>
<td>0.917</td>
<td>0.888</td>
</tr>
<tr>
<td>NNMF</td>
<td>0.919</td>
<td>0.891</td>
</tr>
<tr>
<td>NGCF</td>
<td>0.916</td>
<td>0.896</td>
</tr>
<tr>
<td>PinSAGE</td>
<td>0.923</td>
<td>0.901</td>
</tr>
<tr>
<td>FISM</td>
<td>-</td>
<td>0.752</td>
</tr>
<tr>
<td>MultVAE</td>
<td>-</td>
<td>0.738</td>
</tr>
<tr>
<td><strong>IDCF-NN</strong></td>
<td><strong>0.944</strong></td>
<td><strong>0.939</strong></td>
</tr>
<tr>
<td><strong>IDCF-GC</strong></td>
<td><strong>0.938</strong></td>
<td><strong>0.946</strong></td>
</tr>
</tbody>
</table>

Higher AUC and higher NDCG are better
Further discussions:

- Our model can exceed transductive models w.r.t. RMSE when users' training/historical ratings are sparse.
- There exist informative key users that contribute to most of capacity. Key users with more historical ratings tend to be more important.
- The training time scales linearly w.r.t. dataset size.
Conclusions

- **Our contributions:**
  - propose a new inductive representation model for CF problem
  - guarantee equivalent capacity to MF and can handle new users on-the-fly
  - competitive results on few-shot and SOTA performance on zero-shot users

Thanks for listening!

*Code:* https://github.com/qitianwu/IDCF