Effective Missing Data Prediction for Collaborative Filtering

Hao Ma, Irwin King, and Michael R. Lyu

Department of Computer Science and Engineering
The Chinese University of Hong Kong

SIGIR 2007, Amsterdam, the Netherlands
July 24, 2007
Effective Missing Data Prediction for Collaborative Filtering

1. Introduction
   - Simple Examples of Recommender System
   - Definitions of Some Concepts
   - A Simple CF Example
   - Pearson Correlation Coefficient
   - Significance Weighting

2. Missing Data Prediction
   - Collaborative Filtering Challenges
   - User-Item Matrix
   - Similar Neighbors Selection
   - Missing Data Prediction
   - Parameter Discussion

3. Empirical Analysis
   - Datasets
   - Metrics
   - Summary of Experiments
   - Comparisons
   - Impact of Parameters

4. Conclusions and Future Work
   - Conclusions and Future Work
Effective Missing Data Prediction for Collaborative Filtering

Hao Ma, Irwin King, and Michael R. Lyu

Outline
Introduction
Missing Data Prediction
Empirical Analysis
Conclusions and Future Work

Simple Examples of Recommender System
Definitions of Some Concepts
A Simple CF Example
Pearson Correlation Coefficient
Significance Weighting
Search Using Google

Google

Search: the web pages from Hong Kong

Web

Live Search
Microsoft provides search of the web, news, images and its own encyclopedia, Encarta. Also offers desktop search via a toolbar.

Yahoo!
Welcome to Yahoo!, the world's most visited home page. Quickly find what you're searching for, get in touch with friends and stay in-the-know with the ...

AltaVista
AltaVista provides the most comprehensive search experience on the Web!

MetaCrawler Web Search Home Page
Popular Searches: Online Churches, Blue Book Value, Obituaries, Auto Loan, Airline Tickets, Gift Baskets, See what the world is searching for? ...

MSN.com
MSN's all-in-one Internet portal, the home of Hotmail, MSN Messenger, MSNBC News, Fox Sports, Slate Magazine and more information you care about.

Dogpile Web Search Home Page
Dogpile.com makes searching the Web easy, because it has all the best search engines piled into one. So you get better results from more of the web.

Homepage HotBot Web Search
Offers a search powered by a choice of Google or AskJeeves. There are options to block offensive language, customize search results, and skins.

Hao Ma, Irwin King, and Michael R. Lyu
Effective Missing Data Prediction for Collaborative Filtering
Searching Products on Amazon.com

If a user is viewing the palm Treo 750 Smartphone on Amazon.com, other related information will be recommended to user besides the specification of Treo 750.
If a user is viewing the palm Treo 750 Smartphone on Amazon.com, other related information will be recommended to user besides the specification of Treo 750
Searching Products on Amazon.com

Customers who viewed this item also viewed
- Samsung i607 BlackJack Smartphone (Cingular) by Samsung
- BlackBerry 8100c Pearl (Cingular) by BlackBerry
- Cingular 8525 PDA Phone (Cingular) by HTC
- Sony Ericsson W810i Phone (Cingular) by Sony Ericsson

Customers who bought this item also bought
- PREMIUM RAPID CAR CHARGER for PALM TREO 650 / 680 / 700 / 700w / 700p / 700wx / 750 by Mybat
- Platinum Skin Case w/Swivel Clip --Tre o 650 700w 700p
- OEM 2GB MINISD Mini Secure Digital (SD) Card 2 GB (Bulk Package) by OEM
- palm Treo 680 Smartphone (Cingular) by Palm

These methods are very popular in many online recommendation systems.
Searching Products on Amazon.com

Customers who viewed this item also viewed

- Samsung i607 BlackJack Smartphone (Cingular) by Samsung
- BlackBerry 8100c Pearl (Cingular) by BlackBerry
- Cingular 8525 PDA Phone (Cingular) by HTC
- Sony Ericsson W810i Phone (Cingular) by Sony Ericsson

Customers who bought this item also bought

- PREMIUM RAPID CAR CHARGER for PALM TREO 650 / 680 / 700 / 700w / 700p / 700wx / 750 by Mybat
- Platinum Skin Case w/Swivel Clip -- Treo 650 700w 700p
- OEM 2GB MINISD Mini Secure Digital (SD) Card 2 GB (Bulk Package) by OEM
- palm Treo 680 Smartphone (Cingular) by Palm

- These methods are very popular in many online recommendation systems
More Complicated Recommendations

1. Use the search box above to find your favorite books, movies, albums, artists, authors and brands.
2. Tell us what you think of the items we return for your search by rating the item or telling us you already own them.
3. Repeat until the Recommendations you find in Your Amazon.com reflect your tastes and interests.

Search for items to rate: Amazon.com

Search for items to rate: Music

Search for items to rate: Eniqua
More Complicated Recommendations

Search for items to rate: Music

Search results for Enrique in Music:

1. **Escape**
   ~ Enrique Iglesias
   Your tags: Add (What's this?)
   Rate it
   I Own It

2. **Enrique**
   ~ Enrique Iglesias
   Your tags: Add (What's this?)
   Rate it
   I Own It

3. **Seven**
   ~ Enrique Iglesias
   Your tags: Add (What's this?)
   Rate it
   I Own It
More Complicated Recommendations

Search for items to rate: Music Enrique

Search results for Enrique in Music:

1. Escape
   ~ Enrique Iglesias
   Your tags: Add(What's this?)

2. Enrique
   ~ Enrique Iglesias
   Your tags: Add(What's this?)

3. Seven
   ~ Enrique Iglesias
   Your tags: Add(What's this?)
More Complicated Recommendations

Search results for **Enrique** in Music:

1. **Escape**
   ~ Enrique Iglesias
   **Your tags:** Add (What's this?)
   [Saved] 5 stars
   [Box: I Own It]

2. **Enrique**
   ~ Enrique Iglesias
   **Your tags:** Add (What's this?)
   [Saved] 5 stars
   [Box: I Own It]

3. **Seven**
   ~ Enrique Iglesias
   **Your tags:** Add (What's this?)
   [Saved] 5 stars
   [Box: I Own It]
More Complicated Recommendations

Search for items to rate: Music

Search results for Enrique in Music:

1. **Escape**
   ~ Enrique Iglesias
   Your tags: [Add] (What's this?)

2. **Enrique**
   ~ Enrique Iglesias
   Your tags: [Add] (What's this?)

3. **Seven**
   ~ Enrique Iglesias
   Your tags: [Add] (What's this?)

Five Scales:

★ ★ ★ ★ ★ I hate it
★ ★ ★ ★ I don't like it
★ ★ ★ It's ok
★ ★ I like it
★ I love it
The technique Amazon.com adopts is called Collaborative Filtering!
More Complicated Recommendations

The technique Amazon.com adopts is called Collaborative Filtering!
Google
- Similarity calculation
- Link analysis

Amazon – Simple Example
- User-item matrix is consisted of lots of 0s and 1s
- Frequent pattern mining

Amazon – Complicated Example
- User-item matrix is consisted of lots of ratings which are rated by different users
- Predict other missing data as accurate as possible
<table>
<thead>
<tr>
<th>Outline</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
</tr>
<tr>
<td>Missing Data Prediction</td>
</tr>
<tr>
<td>Empirical Analysis</td>
</tr>
<tr>
<td>Conclusions and Future Work</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Simple Examples of Recommender System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Definitions of Some Concepts</td>
</tr>
<tr>
<td>A Simple CF Example</td>
</tr>
<tr>
<td>Pearson Correlation Coefficient</td>
</tr>
<tr>
<td>Significance Weighting</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Google</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Similarity calculation</td>
</tr>
<tr>
<td>- Link analysis</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Amazon – Simple Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>- User-item matrix is consisted of lots of 0s and 1s</td>
</tr>
<tr>
<td>- Frequent pattern mining</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Amazon – Complicated Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>- User-item matrix is consisted of lots of ratings which are rated by different users</td>
</tr>
<tr>
<td>- Predict other missing data as accurate as possible</td>
</tr>
</tbody>
</table>
Google
- Similarity calculation
- Link analysis

Amazon – Simple Example
- User-item matrix is consisted of lots of 0s and 1s
- Frequent pattern mining

Amazon – Complicated Example
- User-item matrix is consisted of lots of ratings which are rated by different users
- Predict other missing data as accurate as possible
Google

- Similarity calculation
- Link analysis

Amazon – Simple Example

- User-item matrix is consisted of lots of 0s and 1s
- Frequent pattern mining

Amazon – Complicated Example

- User-item matrix is consisted of lots of ratings which are rated by different users
- Predict other missing data as accurate as possible
Definition of Recommendation Systems

- Computer programs
- Predict items that a user may be interested in
- Items could be movies, music, books, news, web pages, etc.
- Given some information about the user’s profile
Definition of Recommendation Systems

- **Computer programs**
- Predict items that a user may be interested in
- Items could be movies, music, books, news, web pages, etc.
- Given some information about the user’s profile
Definition of Recommendation Systems

- Computer programs
- Predict items that a user may be interested in
  - Items could be movies, music, books, news, web pages, etc.
  - Given some information about the user’s profile
Definition of Recommendation Systems

- Computer programs
- Predict items that a user may be interested in
- Items could be movies, music, books, news, web pages, etc.
- Given some information about the user’s profile
Definition of Recommendation Systems

- Computer programs
- Predict items that a user may be interested in
- Items could be movies, music, books, news, web pages, etc.
- Given some information about the user’s profile
Definition of Collaborative Filtering

- Making automatic predictions (filtering) about the interests of a user
- By collecting taste information from many other users (collaborating)
Definition of Collaborative Filtering

- Making automatic predictions (filtering) about the interests of a user
- By collecting taste information from many other users (collaborating)
Definition of Collaborative Filtering

- Making automatic predictions (filtering) about the interests of a user
- By collecting taste information from many other users (collaborating)
User-based Collaborative Filtering

<table>
<thead>
<tr>
<th>Users</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>u₁</td>
</tr>
<tr>
<td>u₁</td>
<td></td>
</tr>
<tr>
<td>u₂</td>
<td>1</td>
</tr>
<tr>
<td>u₃</td>
<td>3</td>
</tr>
<tr>
<td>u₄</td>
<td>3</td>
</tr>
<tr>
<td>u₅</td>
<td>1</td>
</tr>
</tbody>
</table>
### User-based Collaborative Filtering

<table>
<thead>
<tr>
<th>Users</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>u_1</td>
<td>1 3 4 2 5 3 4</td>
</tr>
<tr>
<td>u_2</td>
<td>1 3 4 2 5 3 4</td>
</tr>
<tr>
<td>u_3</td>
<td>3 4 3 4 3 4 4</td>
</tr>
<tr>
<td>u_4</td>
<td>3 4 3 4 3 4 4</td>
</tr>
<tr>
<td>u_5</td>
<td>1 3 5 2 4 1 3</td>
</tr>
<tr>
<td>u_6</td>
<td>1 3 5 2 4 1 3</td>
</tr>
</tbody>
</table>
### User-based Collaborative Filtering

<table>
<thead>
<tr>
<th>Users</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>u₁</td>
<td></td>
</tr>
<tr>
<td>u₂</td>
<td>1</td>
</tr>
<tr>
<td>u₃</td>
<td></td>
</tr>
<tr>
<td>u₄</td>
<td>3</td>
</tr>
<tr>
<td>u₅</td>
<td></td>
</tr>
<tr>
<td>u₆</td>
<td>1</td>
</tr>
</tbody>
</table>

### Definitions of Some Concepts

- **A Simple CF Example**
- **Pearson Correlation Coefficient**
- **Significance Weighting**
### User-based Collaborative Filtering

<table>
<thead>
<tr>
<th>Users</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>u₁</td>
<td>1 3</td>
</tr>
<tr>
<td>u₂</td>
<td>4 2</td>
</tr>
<tr>
<td>u₃</td>
<td>5 3</td>
</tr>
<tr>
<td>u₄</td>
<td>3 4</td>
</tr>
<tr>
<td>u₅</td>
<td>4 4</td>
</tr>
<tr>
<td>u₆</td>
<td>1 3</td>
</tr>
<tr>
<td>u₇</td>
<td>5 2</td>
</tr>
<tr>
<td>u₈</td>
<td>4 1</td>
</tr>
<tr>
<td>u₉</td>
<td>3 3</td>
</tr>
</tbody>
</table>
User-based Collaborative Filtering

Users

<table>
<thead>
<tr>
<th></th>
<th>u₁</th>
<th>u₂</th>
<th>u₃</th>
<th>u₄</th>
<th>u₅</th>
<th>u₆</th>
</tr>
</thead>
<tbody>
<tr>
<td>u₁</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>u₂</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>u₃</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>u₄</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>u₅</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>u₆</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

Items
### User-based Collaborative Filtering

<table>
<thead>
<tr>
<th>Users</th>
<th>Items</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>u₁</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>u₂</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>u₃</td>
<td></td>
<td></td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>u₄</td>
<td>3</td>
<td></td>
<td>4</td>
<td></td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>u₅</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>u₆</td>
<td>1</td>
<td></td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

Hao Ma, Irwin King, and Michael R. Lyu

Effective Missing Data Prediction for Collaborative Filtering
User-based Collaborative Filtering

- User-based collaborative filtering predicts the ratings of active users based on the ratings of similar users found in the user-item matrix.

- The similarity between users could be defined as:

\[
Sim(a, u) = \frac{\sum_{i \in I(a) \cap I(u)} (r_{a,i} - \bar{r}_a) \cdot (r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in I(a) \cap I(u)} (r_{a,i} - \bar{r}_a)^2} \cdot \sqrt{\sum_{i \in I(a) \cap I(u)} (r_{u,i} - \bar{r}_u)^2}}
\]

- \(Sim(a, u)\) is ranging from \([-1, 1]\), and a larger value means users \(a\) and \(u\) are more similar.
User-based Collaborative Filtering

- User-based collaborative filtering predicts the ratings of active users based on the ratings of similar users found in the user-item matrix.

- The similarity between users could be defined as:

\[
Sim(a, u) = \frac{\sum_{i \in I(a) \cap I(u)} (r_{a,i} - \bar{r}_a) \cdot (r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in I(a) \cap I(u)} (r_{a,i} - \bar{r}_a)^2} \cdot \sqrt{\sum_{i \in I(a) \cap I(u)} (r_{u,i} - \bar{r}_u)^2}}
\]

- \(Sim(a, u)\) is ranging from \([-1, 1]\), and a larger value means users \(a\) and \(u\) are more similar.
User-based Collaborative Filtering

- User-based collaborative filtering predicts the ratings of active users based on the ratings of similar users found in the user-item matrix.

- The similarity between users could be defined as:

  \[
  Sim(a, u) = \frac{\sum_{i \in I(a) \cap I(u)} (r_{a,i} - \bar{r}_a) \cdot (r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in I(a) \cap I(u)} (r_{a,i} - \bar{r}_a)^2} \cdot \sqrt{\sum_{i \in I(a) \cap I(u)} (r_{u,i} - \bar{r}_u)^2}}
  \]

- \( Sim(a, u) \) is ranging from \([-1, 1]\), and a larger value means users \( a \) and \( u \) are more similar.
User-based Collaborative Filtering

- User-based collaborative filtering predicts the ratings of active users based on the ratings of similar users found in the user-item matrix.

- The similarity between users could be defined as:

\[
Sim(a, u) = \frac{\sum_{i \in I(a) \cap I(u)} (r_{a,i} - \bar{r}_a) \cdot (r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in I(a) \cap I(u)} (r_{a,i} - \bar{r}_a)^2} \cdot \sqrt{\sum_{i \in I(a) \cap I(u)} (r_{u,i} - \bar{r}_u)^2}}
\]

- \(Sim(a, u)\) is ranging from \([-1, 1]\), and a larger value means users \(a\) and \(u\) are more similar.
### User-based Collaborative Filtering

<table>
<thead>
<tr>
<th>Users</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>u₁</td>
<td></td>
</tr>
<tr>
<td>u₂</td>
<td>4</td>
</tr>
<tr>
<td>u₃</td>
<td></td>
</tr>
<tr>
<td>u₄</td>
<td>2</td>
</tr>
<tr>
<td>u₅</td>
<td></td>
</tr>
<tr>
<td>u₆</td>
<td></td>
</tr>
</tbody>
</table>

Hao Ma, Irwin King, and Michael R. Lyu

Effective Missing Data Prediction for Collaborative Filtering
### User-based Collaborative Filtering

<table>
<thead>
<tr>
<th>Users</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>u₁</td>
<td></td>
</tr>
<tr>
<td>u₂</td>
<td>4</td>
</tr>
<tr>
<td>u₃</td>
<td>4</td>
</tr>
<tr>
<td>u₄</td>
<td>4</td>
</tr>
<tr>
<td>u₅</td>
<td>4</td>
</tr>
<tr>
<td>u₆</td>
<td></td>
</tr>
</tbody>
</table>

The similarity between u₂ and u₄ equals to 1.
Item-based Collaborative Filtering

- Item-based collaborative filtering predicts the ratings of active users based on the information of similar items computed.

- The similarity between items could be defined as:

\[
Sim(i, j) = \frac{\sum_{u \in U(i) \cap U(j)} (r_{u,i} - \bar{r}_i) \cdot (r_{u,j} - \bar{r}_j)}{\sqrt{\sum_{u \in U(i) \cap U(j)} (r_{u,i} - \bar{r}_i)^2} \cdot \sqrt{\sum_{u \in U(i) \cap U(j)} (r_{u,j} - \bar{r}_j)^2}}
\]

- Like user similarity, item similarity \(Sim(i, j)\) is also ranging from \([-1, 1]\).
Item-based Collaborative Filtering

- Item-based collaborative filtering predicts the ratings of active users based on the information of similar items computed.

- The similarity between items could be defined as:

\[
Sim(i, j) = \frac{\sum_{u \in U(i) \cap U(j)} (r_{u,i} - \bar{r}_i) \cdot (r_{u,j} - \bar{r}_j)}{\sqrt{\sum_{u \in U(i) \cap U(j)} (r_{u,i} - \bar{r}_i)^2} \cdot \sqrt{\sum_{u \in U(i) \cap U(j)} (r_{u,j} - \bar{r}_j)^2}}
\]

- Like user similarity, item similarity \(Sim(i, j)\) is also ranging from \([-1, 1]\).
Item-based Collaborative Filtering

- Item-based collaborative filtering predicts the ratings of active users based on the information of similar items computed.

- The similarity between items could be defined as:

\[
Sim(i, j) = \frac{\sum_{u \in U(i) \cap U(j)} (r_{u,i} - \bar{r}_i) \cdot (r_{u,j} - \bar{r}_j)}{\sqrt{\sum_{u \in U(i) \cap U(j)} (r_{u,i} - \bar{r}_i)^2} \cdot \sqrt{\sum_{u \in U(i) \cap U(j)} (r_{u,j} - \bar{r}_j)^2}}
\]

- Like user similarity, item similarity $Sim(i, j)$ is also ranging from $[-1, 1]$. 

Hao Ma, Irwin King, and Michael R. Lyu  Effective Missing Data Prediction for Collaborative Filtering
Item-based Collaborative Filtering

- Item-based collaborative filtering predicts the ratings of active users based on the information of similar items computed.
- The similarity between items could be defined as:

\[
Sim(i, j) = \frac{\sum_{u \in U(i) \cap U(j)} (r_{u,i} - \bar{r}_i) \cdot (r_{u,j} - \bar{r}_j)}{\sqrt{\sum_{u \in U(i) \cap U(j)} (r_{u,i} - \bar{r}_i)^2} \cdot \sqrt{\sum_{u \in U(i) \cap U(j)} (r_{u,j} - \bar{r}_j)^2}}
\]

- Like user similarity, item similarity \( Sim(i, j) \) is also ranging from \([-1, 1]\).
## An Example

<table>
<thead>
<tr>
<th>Users</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 3 2 5 3 2 3</td>
<td>3 2 1 5 4 1 4</td>
</tr>
</tbody>
</table>
### An Example

<table>
<thead>
<tr>
<th>Users</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
</tr>
</tbody>
</table>
An Example

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

Item 3

Do these two users really have the same taste???
**Significance Weighting**

- We use the following equation to solve this problem:

\[
Sim'(a, u) = \frac{\min(|I_a \cap I_u|, \gamma)}{\gamma} \cdot Sim(a, u),
\]

where \(|I_a \cap I_u|\) is the number of items which user \(a\) and user \(u\) rated in common.

- Then the similarity between items could be defined as:

\[
Sim'(i, j) = \frac{\min(|U_i \cap U_j|, \delta)}{\delta} \cdot Sim(i, j),
\]

where \(|U_i \cap U_j|\) is the number of users who rated both item \(i\) and item \(j\).
Significance Weighting

We use the following equation to solve this problem:

\[
Sim'(a, u) = \frac{\min(|I_a \cap I_u|, \gamma)}{\gamma} \cdot Sim(a, u),
\]

where \( |I_a \cap I_u| \) is the number of items which user \( a \) and user \( u \) rated in common.

Then the similarity between items could be defined as:

\[
Sim'(i, j) = \frac{\min(|U_i \cap U_j|, \delta)}{\delta} \cdot Sim(i, j),
\]

where \( |U_i \cap U_j| \) is the number of users who rated both item \( i \) and item \( j \).
Significance Weighting

- We use the following equation to solve this problem:

\[
Sim'(a, u) = \frac{\text{Min}(|I_a \cap I_u|, \gamma)}{\gamma} \cdot Sim(a, u),
\]

where \( |I_a \cap I_u| \) is the number of items which user \( a \) and user \( u \) rated in common.

- Then the similarity between items could be defined as:

\[
Sim'(i, j) = \frac{\text{Min}(|U_i \cap U_j|, \delta)}{\delta} \cdot Sim(i, j),
\]

where \( |U_i \cap U_j| \) is the number of users who rated both item \( i \) and item \( j \).
Challenges of Collaborative Filtering

- Data Sparsity
- Prediction Accuracy
- Scalability
User-Item Matrix

<table>
<thead>
<tr>
<th></th>
<th>$i_1$</th>
<th>$i_2$</th>
<th>$i_3$</th>
<th>$i_4$</th>
<th>$i_5$</th>
<th>$i_6$</th>
<th>$i_7$</th>
<th>$i_8$</th>
<th>$i_9$</th>
<th>$i_n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>$r_{1,1}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$u_2$</td>
<td></td>
<td>$r_{2,2}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$u_3$</td>
<td></td>
<td></td>
<td>$r_{3,3}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$u_4$</td>
<td></td>
<td></td>
<td></td>
<td>$r_{4,4}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$u_5$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$r_{5,5}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$u_6$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$r_{6,6}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$u_m$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$r_{m,m}$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(a)

Challenges of Collaborative Filtering

- Data Sparsity
- Prediction Accuracy
- Scalability
User-Item Matrix

Challenges of Collaborative Filtering

- Data Sparsity
- Prediction Accuracy
- Scalability
### Collaborative Filtering Challenges

- **User-Item Matrix**
- **Similar Neighbors Selection**
- **Missing Data Prediction**
- **Parameter Discussion**

### User-Item Matrix

![User-Item Matrix Diagram](image)

### Challenges of Collaborative Filtering

- **Data Sparsity**
- **Prediction Accuracy**
- **Scalability**
Collaborative Filtering Challenges

- User-Item Matrix
- Similar Neighbors Selection
- Missing Data Prediction
- Parameter Discussion

User-Item Matrix

\[
\begin{array}{cccccccc}
  & i_1 & i_2 & i_3 & i_4 & i_5 & i_6 & i_7 & i_8 & i_9 & i_n \\
 u_1 & r_{1,1} & r_{1,3} & r_{1,4} & & & & & & & \\
m_{2} & r_{2,2} & r_{2,3} & r_{2,4} & & & & & & & \\
m_{3} & & r_{3,3} & r_{3,4} & & & & & & & \\
m_{4} & & & r_{4,4} & & & & & & & \\
m_{5} & & & & r_{5,5} & & & & & & \\
m_{6} & & & & & r_{6,6} & & & & & \\
m_{m} & & & & & & r_{m,7} & & & & \\
  & & & & & & & r_{m,8} & & & \\
  & & & & & & & & r_{m,n} & & \\
\end{array}
\]

Challenges of Collaborative Filtering

- Data Sparsity
- Prediction Accuracy
- Scalability
Challenges of Collaborative Filtering

- Data Sparsity
- Prediction Accuracy
- Scalability

Data Sparsity
- Propose an algorithm to increase the density of User-Item Matrix
- Only predict some of the missing data

Prediction Accuracy
- Adopt significance weighting
- Linearly combine user information with item information
- Predict the missing data with high confidence
- Our algorithm increases 6.24% of prediction accuracy over other state-of-the-art methods in average
Challenges of Collaborative Filtering

- Data Sparsity
- Prediction Accuracy
- Scalability

Data Sparsity
- Propose an algorithm to increase the density of User-Item Matrix
- Only predict some of the missing data

Prediction Accuracy
- Adopt significance weighting
- Linearly combine user information with item information
- Predict the missing data with high confidence
- Our algorithm increases 6.24% of prediction accuracy over other state-of-the-art methods in average
Challenges of Collaborative Filtering

- Data Sparsity
- Prediction Accuracy
- Scalability

Data Sparsity

- Propose an algorithm to increase the density of User-Item Matrix
- Only predict some of the missing data

Prediction Accuracy

- Adopt significance weighting
- Linearly combine user information with item information
- Predict the missing data with high confidence
- Our algorithm increases 6.24% of prediction accuracy over other state-of-the-art methods in average
Challenges of Collaborative Filtering

- Data Sparsity
- Prediction Accuracy
- Scalability

Data Sparsity

- Propose an algorithm to increase the density of User-Item Matrix
- Only predict some of the missing data

Prediction Accuracy

- Adopt significance weighting
- Linearly combine user information with item information
- Predict the missing data with high confidence
- Our algorithm increases 6.24% of prediction accuracy over other state-of-the-art methods in average
Challenges of Collaborative Filtering

- Data Sparsity
- Prediction Accuracy
- Scalability

Data Sparsity

- Propose an algorithm to increase the density of User-Item Matrix
- Only predict some of the missing data

Prediction Accuracy

- Adopt significance weighting
- Linearly combine user information with item information
- Predict the missing data with high confidence
- Our algorithm increases 6.24% of prediction accuracy over other state-of-the-art methods on average
Challenges of Collaborative Filtering

- Data Sparsity
- Prediction Accuracy
- Scalability

Data Sparsity

- Propose an algorithm to increase the density of User-Item Matrix
- Only predict some of the missing data

Prediction Accuracy

- Adopt significance weighting
  - Linearly combine user information with item information
  - Predict the missing data with high confidence
  - Our algorithm increases 6.24% of prediction accuracy over other state-of-the-art methods in average
Challenges of Collaborative Filtering

- Data Sparsity
- Prediction Accuracy
- Scalability

Data Sparsity

- Propose an algorithm to increase the density of User-Item Matrix
- Only predict some of the missing data

Prediction Accuracy

- Adopt significance weighting
- Linearly combine user information with item information
- Predict the missing data with high confidence
- Our algorithm increases $6.24\%$ of prediction accuracy over other state-of-the-art methods in average
Challenges of Collaborative Filtering

- Data Sparsity
- Prediction Accuracy
- Scalability

Data Sparsity

- Propose an algorithm to increase the density of User-Item Matrix
- Only predict some of the missing data

Prediction Accuracy

- Adopt significance weighting
- Linearly combine user information with item information
- Predict the missing data with high confidence
- Our algorithm increases 6.24% of prediction accuracy over other state-of-the-art methods in average
**Challenges of Collaborative Filtering**

- **Data Sparsity**
- **Prediction Accuracy**
- **Scalability**

**Data Sparsity**

- Propose an algorithm to increase the density of User-Item Matrix
- Only predict some of the missing data

**Prediction Accuracy**

- Adopt significance weighting
- Linearly combine user information with item information
- Predict the missing data with high confidence
- Our algorithm increases 6.24% of prediction accuracy over other state-of-the-art methods in average
Collaborative Filtering Challenges

User-Item Matrix

Similar Neighbors Selection

Missing Data Prediction

Parameter Discussion

Hao Ma, Irwin King, and Michael R. Lyu

Effective Missing Data Prediction for Collaborative Filtering
Similar Neighbors Selection

- For every missing data $r_{u,i}$, a set of similar users $S(u)$ towards user $u$ can be generated according to:

$$S(u) = \{u_a | Sim'(u_a, u) > \eta, u_a \neq u\}$$

where $Sim'(u_a, u)$ is computed using Significance Weighting, and $\eta$ is the user similarity threshold.

- At the same time, for every missing data $r_{u,i}$, a set of similar items $S(i)$ towards item $i$ can be generated according to:

$$S(i) = \{i_k | Sim'(i_k, i) > \theta, i_k \neq i\}$$

where $\theta$ is the item similarity threshold.
Similar Neighbors Selection

- For every missing data $r_{u,i}$, a set of similar users $S(u)$ towards user $u$ can be generated according to:

$$S(u) = \{ u_a | Sim'(u_a, u) > \eta, u_a \neq u \}$$

where $Sim'(u_a, u)$ is computed using Significance Weighting, and $\eta$ is the user similarity threshold.

- At the same time, for every missing data $r_{u,i}$, a set of similar items $S(i)$ towards item $i$ can be generated according to:

$$S(i) = \{ i_k | Sim'(i_k, i) > \theta, i_k \neq i \}$$

where $\theta$ is the item similarity threshold.
Similar Neighbors Selection

- For every missing data $r_{u,i}$, a set of similar users $S(u)$ towards user $u$ can be generated according to:

$$S(u) = \{u_a | Sim'(u_a, u) > \eta, u_a \neq u\}$$

where $Sim'(u_a, u)$ is computed using Significance Weighting, and $\eta$ is the user similarity threshold.

- At the same time, for every missing data $r_{u,i}$, a set of similar items $S(i)$ towards item $i$ can be generated according to:

$$S(i) = \{i_k | Sim'(i_k, i) > \theta, i_k \neq i\}$$

where $\theta$ is the item similarity threshold.
Missing Data Prediction Algorithm

Given the missing data $r_{u,i}$, if $S(u) \neq \emptyset \land S(i) \neq \emptyset$, the prediction of missing data $P(r_{u,i})$ is defined as:

$$P(r_{u,i}) = \lambda \times (\bar{u} + \frac{\sum_{u_a \in S(u)} Sim'(u_a, u) \cdot (r_{u_a,i} - \bar{u}_a)}{\sum_{u_a \in S(u)} Sim'(u_a, u)}) +$$

$$+ (1 - \lambda) \times (\bar{i} + \frac{\sum_{i_k \in S(i)} Sim'(i_k, i) \cdot (r_{u,i_k} - \bar{i}_k)}{\sum_{i_k \in S(i)} Sim'(i_k, i)})$$
Given the missing data $r_{u,i}$, if $S(u) \neq \emptyset \land S(i) \neq \emptyset$, the prediction of missing data $P(r_{u,i})$ is defined as:

$$P(r_{u,i}) = \lambda \times (\bar{u} + \frac{\sum_{u_{a} \in S(u)} Sim'(u_{a}, u) \cdot (r_{u_{a},i} - \bar{u}_{a})}{\sum_{u_{a} \in S(u)} Sim'(u_{a}, u)}) +$$

$$(1 - \lambda) \times (\bar{i} + \frac{\sum_{i_{k} \in S(i)} Sim'(i_{k}, i) \cdot (r_{u,i_{k}} - \bar{i}_{k})}{\sum_{i_{k} \in S(i)} Sim'(i_{k}, i)})$$
**Missing Data Prediction Algorithm**

- If $S(u) \neq \emptyset \land S(i) = \emptyset$, the prediction of missing data $P(r_{u,i})$ is defined as:

$$
P(r_{u,i}) = \bar{u} + \sum_{u_a \in S(u)} \frac{Sim'(u_a, u) \cdot (r_{u_a,i} - \bar{u}_a)}{\sum_{u_a \in S(u)} Sim'(u_a, u)}
$$

- If $S(u) = \emptyset \land S(i) \neq \emptyset$, the prediction of missing data $P(r_{u,i})$ is defined as:

$$
P(r_{u,i}) = \bar{i} + \sum_{i_k \in S(i)} \frac{Sim'(i_k, i) \cdot (r_{u,i_k} - \bar{i}_k)}{\sum_{i_k \in S(i)} Sim'(i_k, i)}
$$
Missing Data Prediction Algorithm

- If $S(u) \neq \emptyset \land S(i) = \emptyset$, the prediction of missing data $P(r_u,i)$ is defined as:

\[
P(r_u,i) = \bar{u} + \frac{\sum_{u_a \in S(u)} Sim'(u_a,u) \cdot (r_{u_a,i} - \bar{u}_a)}{\sum_{u_a \in S(u)} Sim'(u_a,u)}
\]

- If $S(u) = \emptyset \land S(i) \neq \emptyset$, the prediction of missing data $P(r_u,i)$ is defined as:

\[
P(r_u,i) = \bar{i} + \frac{\sum_{i_k \in S(i)} Sim'(i_k,i) \cdot (r_{u,i_k} - \bar{i}_k)}{\sum_{i_k \in S(i)} Sim'(i_k,i)}
\]
If $S(u) \neq \emptyset \land S(i) = \emptyset$, the prediction of missing data $P(r_{u,i})$ is defined as:

$$P(r_{u,i}) = \bar{u} + \frac{\sum_{u_a \in S(u)} Sim'(u_a, u) \cdot (r_{u_a,i} - \bar{u}_a)}{\sum_{u_a \in S(u)} Sim'(u_a, u)}$$

If $S(u) = \emptyset \land S(i) \neq \emptyset$, the prediction of missing data $P(r_{u,i})$ is defined as:

$$P(r_{u,i}) = \bar{i} + \frac{\sum_{i_k \in S(i)} Sim'(i_k, i) \cdot (r_{u,i_k} - \bar{i}_k)}{\sum_{i_k \in S(i)} Sim'(i_k, i)}$$
If $S(u) = \emptyset \land S(i) = \emptyset$, the prediction of missing data $P(r_{u,i})$ is defined as:

$$P(r_{u,i}) = 0$$

This consideration is different from all other existing prediction or smoothing methods – they always try to predict all the missing data in the user-item matrix, which will predict some missing data with bad quality.
If \( S(u) = \emptyset \land S(i) = \emptyset \), the prediction of missing data \( P(r_{u,i}) \) is defined as:

\[
P(r_{u,i}) = 0
\]

This consideration is different from all other existing prediction or smoothing methods – they always try to predict all the missing data in the user-item matrix, which will predict some missing data with bad quality.
If \( S(u) = \emptyset \wedge S(i) = \emptyset \), the prediction of missing data \( P(r_{u,i}) \) is defined as:

\[
P(r_{u,i}) = 0
\]

This consideration is different from all other existing prediction or smoothing methods – they always try to predict all the missing data in the user-item matrix, which will predict some missing data with bad quality.
Discussion on $\gamma$ and $\delta$

- Employed to avoid overestimating the user similarities and item similarities
- Too high $\Rightarrow$ users or items do not have enough neighbors $\Rightarrow$ decrease of prediction accuracy
- Too low $\Rightarrow$ overestimate problem still exists $\Rightarrow$ decrease of prediction accuracy
**Discussion on $\gamma$ and $\delta$**

- Employed to avoid overestimating the user similarities and item similarities
  - Too high $\Rightarrow$ users or items do not have enough neighbors $\Rightarrow$ decrease of prediction accuracy
  - Too low $\Rightarrow$ overestimate problem still exists $\Rightarrow$ decrease of prediction accuracy
Discussion on $\gamma$ and $\delta$

- Employed to avoid overestimating the user similarities and item similarities
- Too high $\Rightarrow$ users or items do not have enough neighbors $\Rightarrow$ decrease of prediction accuracy
- Too low $\Rightarrow$ overestimate problem still exists $\Rightarrow$ decrease of prediction accuracy
Discussion on $\gamma$ and $\delta$

- Employed to avoid overestimating the user similarities and item similarities
- Too high $\Rightarrow$ users or items do not have enough neighbors $\Rightarrow$ decrease of prediction accuracy
- Too low $\Rightarrow$ overestimate problem still exists $\Rightarrow$ decrease of prediction accuracy
Parameter Discussion on $\eta$ and $\theta$

- Thresholds to select neighbors
- Too high $\Rightarrow$ few missing data need to be predicted $\Rightarrow$ user-item matrix is very sparse
- Too low $\Rightarrow$ almost all the missing data need to be predicted $\Rightarrow$ user-item matrix is very dense
Discussion on $\eta$ and $\theta$

- **Thresholds to select neighbors**
  - Too high $\Rightarrow$ few missing data need to be predicted $\Rightarrow$ user-item matrix is very sparse
  - Too low $\Rightarrow$ almost all the missing data need to be predicted $\Rightarrow$ user-item matrix is very dense
Discussion on $\eta$ and $\theta$

- Thresholds to select neighbors
- Too high $\implies$ few missing data need to be predicted $\implies$ user-item matrix is very sparse
- Too low $\implies$ almost all the missing data need to be predicted $\implies$ user-item matrix is very dense
Collaborative Filtering Challenges
User-Item Matrix
Similar Neighbors Selection
Missing Data Prediction

Parameter Discussion

- **Discussion on $\eta$ and $\theta$**
  - Thresholds to select neighbors
  - Too high $\Rightarrow$ few missing data need to be predicted $\Rightarrow$ user-item matrix is very sparse
  - Too low $\Rightarrow$ almost all the missing data need to be predicted $\Rightarrow$ user-item matrix is very dense

Hao Ma, Irwin King, and Michael R. Lyu

Effective Missing Data Prediction for Collaborative Filtering
Discussion on $\lambda$

- Determines how closely the rating prediction relies on user information or item information
- $\lambda = 1 \implies$ prediction depends completely upon user-based information
- $\lambda = 0 \implies$ prediction depends completely upon item-based information
Parameter
- $\gamma$
- $\delta$
- $\eta$
- $\theta$
- $\lambda$

Discussion on $\lambda$
- Determines how closely the rating prediction relies on user information or item information
- $\lambda = 1 \implies$ prediction depends completely upon user-based information
- $\lambda = 0 \implies$ prediction depends completely upon item-based information
Parameter Discussion on $\lambda$

- Determines how closely the rating prediction relies on user information or item information
- $\lambda = 1 \implies$ prediction depends completely upon user-based information
- $\lambda = 0 \implies$ prediction depends completely upon item-based information
Discussion on $\lambda$

- Determines how closely the rating prediction relies on user information or item information
- $\lambda = 1 \implies$ prediction depends completely upon user-based information
- $\lambda = 0 \implies$ prediction depends completely upon item-based information
Parameter Discussion

Table: The relationship between parameters with other CF approaches (MDP: Mission Data Predicted)

<table>
<thead>
<tr>
<th>λ</th>
<th>η</th>
<th>θ</th>
<th>Related CF Approaches</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>User-based CF without MDP</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>Item-based CF without MDP</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>User-based CF with full MDP</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Item-based CF with full MDP</td>
</tr>
</tbody>
</table>
Parameter Discussion

**Table:** The relationship between parameters with other CF approaches (MDP: Mission Data Predicted)

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>$\eta$</th>
<th>$\theta$</th>
<th><strong>Related CF Approaches</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>User-based CF without MDP</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>Item-based CF without MDP</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>User-based CF with full MDP</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Item-based CF with full MDP</td>
</tr>
</tbody>
</table>
Movielens

- It contains 100,000 ratings (1-5 scales) rated by 943 users on 1,682 movies, and each user at least rated 20 movies. The density of the user-item matrix is:

\[
\frac{100000}{943 \times 1682} = 6.30\%
\]

- The statistics of dataset MovieLens is summarized in the following table:

<table>
<thead>
<tr>
<th>Statistics</th>
<th>User</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. Num. of Ratings</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>Max. Num. of Ratings</td>
<td>737</td>
<td>583</td>
</tr>
<tr>
<td>Avg. Num. of Ratings</td>
<td>106.04</td>
<td>59.45</td>
</tr>
</tbody>
</table>
Movielens

- It contains 100,000 ratings (1-5 scales) rated by 943 users on 1,682 movies, and each user at least rated 20 movies. The density of the user-item matrix is:

\[
\frac{100,000}{943 \times 1682} = 6.30\%
\]

- The statistics of dataset MovieLens is summarized in the following table:

<table>
<thead>
<tr>
<th>Statistics</th>
<th>User</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. Num. of Ratings</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>Max. Num. of Ratings</td>
<td>737</td>
<td>583</td>
</tr>
<tr>
<td>Avg. Num. of Ratings</td>
<td>106.04</td>
<td>59.45</td>
</tr>
</tbody>
</table>
Movielens

* It contains **100,000** ratings (1-5 scales) rated by **943** users on **1,682** movies, and each user at least rated 20 movies. The density of the user-item matrix is:

\[
\frac{100000}{943 \times 1682} = 6.30\%
\]

* The statistics of dataset MovieLens is summarized in the following table:

<table>
<thead>
<tr>
<th>Statistics</th>
<th>User</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. Num. of Ratings</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>Max. Num. of Ratings</td>
<td>737</td>
<td>583</td>
</tr>
<tr>
<td>Avg. Num. of Ratings</td>
<td>106.04</td>
<td>59.45</td>
</tr>
</tbody>
</table>
Mean Absolute Errors

- We use the Mean Absolute Error (MAE) metrics to measure the prediction quality of our proposed approach with other collaborative filtering methods.

- MAE is defined as:

\[
MAE = \frac{\sum_{u,i} |r_{u,i} - \hat{r}_{u,i}|}{N},
\]

where \(r_{u,i}\) denotes the rating that user \(u\) gave to item \(i\), and \(\hat{r}_{u,i}\) denotes the rating that user \(u\) gave to item \(i\) which is predicted by our approach, and \(N\) denotes the number of tested ratings.
**Mean Absolute Errors**

- We use the Mean Absolute Error (MAE) metrics to measure the prediction quality of our proposed approach with other collaborative filtering methods.

MAE is defined as:

\[
MAE = \frac{\sum_{u,i} |r_{u,i} - \hat{r}_{u,i}|}{N},
\]

where \(r_{u,i}\) denotes the rating that user \(u\) gave to item \(i\), and \(\hat{r}_{u,i}\) denotes the rating that user \(u\) gave to item \(i\) which is predicted by our approach, and \(N\) denotes the number of tested ratings.
Mean Absolute Errors

- We use the Mean Absolute Error (MAE) metrics to measure the prediction quality of our proposed approach with other collaborative filtering methods.
- MAE is defined as:

\[
MAE = \frac{\sum_{u,i} |r_{u,i} - \hat{r}_{u,i}|}{N},
\]

where \( r_{u,i} \) denotes the rating that user \( u \) gave to item \( i \), and \( \hat{r}_{u,i} \) denotes the rating that user \( u \) gave to item \( i \) which is predicted by our approach, and \( N \) denotes the number of tested ratings.
Summary of Experiments

- Comparisons with Traditional PCC Methods
- Comparisons with State-of-the-Art Algorithms
- Impact of Missing Data Prediction
- Impact of $\gamma$ and $\delta$
- Impact of $\lambda$
- Impact of $\eta$ and $\theta$
Summary of Experiments

- Comparisons with Traditional PCC Methods
- Comparisons with State-of-the-Art Algorithms
- Impact of Missing Data Prediction
- Impact of $\gamma$ and $\delta$
- Impact of $\lambda$
- Impact of $\eta$ and $\theta$

Comparisons with Traditional PCC Methods

- User-based collaborative filtering using Pearson Correlation Coefficient
- Item-based collaborative filtering using Pearson Correlation Coefficient
Summary of Experiments

- Comparisons with Traditional PCC Methods
- **Comparisons with State-of-the-Art Algorithms**
- Impact of Missing Data Prediction
- Impact of $\gamma$ and $\delta$
- Impact of $\lambda$
- Impact of $\eta$ and $\theta$

Comparisons with State-of-the-Art Algorithms

- Similarity Fusion (SF) [J. Wang, et al., SIGIR 2006]
- Smoothing and Cluster-Based PCC (SCBPCC) [G. Xue, et al., SIGIR 2005]
- Aspect Model (AM) [T. Hofmann, TOIS 2004]
- Personality Diagnosis (PD) [D. M. Pennock, et al., UAI 2000]
Summary of Experiments

- Comparisons with Traditional PCC Methods
- Comparisons with State-of-the-Art Algorithms
- **Impact of Missing Data Prediction**
- Impact of $\gamma$ and $\delta$
- Impact of $\lambda$
- Impact of $\eta$ and $\theta$

Impact of Missing Data Prediction

- Effective Missing Data Prediction (EMDP)
- Predict Every Missing Data (PEMD)
Summary of Experiments

- Comparisons with Traditional PCC Methods
- Comparisons with State-of-the-Art Algorithms
- Impact of Missing Data Prediction
- Impact of $\gamma$ and $\delta$
- Impact of $\lambda$
- Impact of $\eta$ and $\theta$

Impact of Parameters

- Impact of each parameter
### MAE Comparisons with PCC Methods

**Table:** MAE comparison with other approaches (A smaller MAE value means a better performance)

<table>
<thead>
<tr>
<th>Training Users</th>
<th>Methods</th>
<th>Given5</th>
<th>Given10</th>
<th>Given20</th>
</tr>
</thead>
<tbody>
<tr>
<td>MovieLens 300</td>
<td>EMDP</td>
<td>0.784</td>
<td>0.765</td>
<td>0.755</td>
</tr>
<tr>
<td></td>
<td>UPCC</td>
<td>0.838</td>
<td>0.814</td>
<td>0.802</td>
</tr>
<tr>
<td></td>
<td>IPCC</td>
<td>0.870</td>
<td>0.838</td>
<td>0.813</td>
</tr>
<tr>
<td>MovieLens 200</td>
<td>EMDP</td>
<td>0.796</td>
<td>0.770</td>
<td>0.761</td>
</tr>
<tr>
<td></td>
<td>UPCC</td>
<td>0.843</td>
<td>0.822</td>
<td>0.807</td>
</tr>
<tr>
<td></td>
<td>IPCC</td>
<td>0.855</td>
<td>0.834</td>
<td>0.812</td>
</tr>
<tr>
<td>MovieLens 100</td>
<td>EMDP</td>
<td>0.811</td>
<td>0.778</td>
<td>0.769</td>
</tr>
<tr>
<td></td>
<td>UPCC</td>
<td>0.876</td>
<td>0.847</td>
<td>0.811</td>
</tr>
<tr>
<td></td>
<td>IPCC</td>
<td>0.890</td>
<td>0.850</td>
<td>0.824</td>
</tr>
</tbody>
</table>
#### MAE Comparisons with PCC Methods

Table: MAE comparison with other approaches (A smaller MAE value means a better performance)

<table>
<thead>
<tr>
<th>Training Users</th>
<th>Methods</th>
<th>Given5</th>
<th>Given10</th>
<th>Given20</th>
</tr>
</thead>
<tbody>
<tr>
<td>MovieLens 300</td>
<td>EMDP</td>
<td>0.784</td>
<td>0.765</td>
<td>0.755</td>
</tr>
<tr>
<td></td>
<td>UPCC</td>
<td>0.838</td>
<td>0.814</td>
<td>0.802</td>
</tr>
<tr>
<td></td>
<td>IPCC</td>
<td>0.870</td>
<td>0.838</td>
<td>0.813</td>
</tr>
<tr>
<td>MovieLens 200</td>
<td>EMDP</td>
<td><strong>0.796</strong></td>
<td><strong>0.770</strong></td>
<td><strong>0.761</strong></td>
</tr>
<tr>
<td></td>
<td>UPCC</td>
<td>0.843</td>
<td>0.822</td>
<td>0.807</td>
</tr>
<tr>
<td></td>
<td>IPCC</td>
<td>0.855</td>
<td>0.834</td>
<td>0.812</td>
</tr>
<tr>
<td>MovieLens 100</td>
<td>EMDP</td>
<td><strong>0.811</strong></td>
<td><strong>0.778</strong></td>
<td><strong>0.769</strong></td>
</tr>
<tr>
<td></td>
<td>UPCC</td>
<td>0.876</td>
<td>0.847</td>
<td>0.811</td>
</tr>
<tr>
<td></td>
<td>IPCC</td>
<td>0.890</td>
<td>0.850</td>
<td>0.824</td>
</tr>
</tbody>
</table>
## MAE Comparisons with State-of-the-Art Algorithms

Table: MAE comparison with state-of-the-art algorithms (A smaller MAE value means a better performance)

<table>
<thead>
<tr>
<th>Num. of Training Users</th>
<th>100</th>
<th></th>
<th>200</th>
<th></th>
<th>300</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Ratings Given</td>
<td>5</td>
<td>10</td>
<td>20</td>
<td>5</td>
<td>10</td>
<td>20</td>
</tr>
<tr>
<td>EMDP</td>
<td>0.807</td>
<td>0.769</td>
<td>0.765</td>
<td>0.793</td>
<td>0.760</td>
<td>0.751</td>
</tr>
<tr>
<td>SF</td>
<td>0.847</td>
<td>0.774</td>
<td>0.792</td>
<td>0.827</td>
<td>0.773</td>
<td>0.783</td>
</tr>
<tr>
<td>SCBPCC</td>
<td>0.848</td>
<td>0.819</td>
<td>0.789</td>
<td>0.831</td>
<td>0.813</td>
<td>0.784</td>
</tr>
<tr>
<td>AM</td>
<td>0.963</td>
<td>0.922</td>
<td>0.887</td>
<td>0.849</td>
<td>0.837</td>
<td>0.815</td>
</tr>
<tr>
<td>PD</td>
<td>0.849</td>
<td>0.817</td>
<td>0.808</td>
<td>0.836</td>
<td>0.815</td>
<td>0.792</td>
</tr>
<tr>
<td>PCC</td>
<td>0.874</td>
<td>0.836</td>
<td>0.818</td>
<td>0.859</td>
<td>0.829</td>
<td>0.813</td>
</tr>
</tbody>
</table>

Hao Ma, Irwin King, and Michael R. Lyu
Effective Missing Data Prediction for Collaborative Filtering
### MAE Comparisons with State-of-the-Art Algorithms

**Table:** MAE comparison with state-of-the-art algorithms (A smaller MAE value means a better performance)

<table>
<thead>
<tr>
<th>Num. of Training Users</th>
<th>5</th>
<th>10</th>
<th>20</th>
<th>5</th>
<th>10</th>
<th>20</th>
<th>5</th>
<th>10</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ratings Given</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMDP</td>
<td>0.807</td>
<td>0.769</td>
<td>0.765</td>
<td>0.793</td>
<td>0.760</td>
<td>0.751</td>
<td>0.788</td>
<td>0.754</td>
<td>0.746</td>
</tr>
<tr>
<td>SF</td>
<td>0.847</td>
<td>0.774</td>
<td>0.792</td>
<td>0.827</td>
<td>0.773</td>
<td>0.783</td>
<td>0.804</td>
<td>0.761</td>
<td>0.769</td>
</tr>
<tr>
<td>SCBPCC</td>
<td>0.848</td>
<td>0.819</td>
<td>0.789</td>
<td>0.831</td>
<td>0.813</td>
<td>0.784</td>
<td>0.822</td>
<td>0.810</td>
<td>0.778</td>
</tr>
<tr>
<td>AM</td>
<td>0.963</td>
<td>0.922</td>
<td>0.887</td>
<td>0.849</td>
<td>0.837</td>
<td>0.815</td>
<td>0.820</td>
<td>0.822</td>
<td>0.796</td>
</tr>
<tr>
<td>PD</td>
<td>0.849</td>
<td>0.817</td>
<td>0.808</td>
<td>0.836</td>
<td>0.815</td>
<td>0.792</td>
<td>0.827</td>
<td>0.815</td>
<td>0.789</td>
</tr>
<tr>
<td>PCC</td>
<td>0.874</td>
<td>0.836</td>
<td>0.818</td>
<td>0.859</td>
<td>0.829</td>
<td>0.813</td>
<td>0.849</td>
<td>0.841</td>
<td>0.820</td>
</tr>
</tbody>
</table>
Impact of Missing Data Prediction

Figure: MAE Comparison of EMDP and PEMD (A smaller MAE value means a better performance)
Impact of $\gamma$ and $\delta$

Figure: Impact of $\gamma$ and $\delta$ on MAE and Matrix Density
**Impact of $\lambda$**

**Figure:** Impact of $\lambda$ on MAE
Impact of $\eta$ and $\theta$ on MAE and Density

Figure: Impact of $\eta$ and $\theta$ on MAE and Density
Conclusions

- Proposes an **effective missing data prediction algorithm** for Collaborative Filtering
- **Combines** users information and items information together
- **Outperforms** other state-of-the-art collaborative filtering approaches

Future Work

- Explore the **relationship** between user information and item information
- **Scalability** analysis and improvement of our algorithm
- Employ more **metrics** to measure our algorithm
Conclusions

- Proposes an effective missing data prediction algorithm for Collaborative Filtering
- Combines users information and items information together
- Outperforms other state-of-the-art collaborative filtering approaches

Future Work

- Explore the relationship between user information and item information
- Scalability analysis and improvement of our algorithm
- Employ more metrics to measure our algorithm
Conclusions

- Proposes an **effective missing data prediction algorithm** for Collaborative Filtering
- **Combines** users information and items information together
- **Outperforms** other state-of-the-art collaborative filtering approaches

Future Work

- Explore the relationship between user information and item information
- Scalability analysis and improvement of our algorithm
- Employ more metrics to measure our algorithm
Conclusions

- Proposes an **effective missing data prediction algorithm** for Collaborative Filtering
- **Combines** users information and items information together
- **Outperforms** other state-of-the-art collaborative filtering approaches

Future Work

- Explore the **relationship** between user information and item information
- **Scalability analysis and improvement** of our algorithm
- Employ more metrics to measure our algorithm
Conclusions

- Proposes an **effective missing data prediction algorithm** for Collaborative Filtering
- **Combines** users information and items information together
- **Outperforms** other state-of-the-art collaborative filtering approaches

Future Work

- Explore the **relationship** between user information and item information
- **Scalability analysis and improvement** of our algorithm
- Employ more metrics to measure our algorithm
Conclusions

- Proposes an effective missing data prediction algorithm for Collaborative Filtering
- Combines users information and items information together
- Outperforms other state-of-the-art collaborative filtering approaches

Future Work

- Explore the relationship between user information and item information
- Scalability analysis and improvement of our algorithm
- Employ more metrics to measure our algorithm
Conclusions

- Proposes an **effective missing data prediction algorithm** for Collaborative Filtering
- **Combines** users information and items information together
- **Outperforms** other state-of-the-art collaborative filtering approaches

Future Work

- Explore the **relationship** between user information and item information
- **Scalability analysis and improvement** of our algorithm
- Employ more **metrics** to measure our algorithm
Conclusions

- Proposes an effective missing data prediction algorithm for Collaborative Filtering
- Combines users information and items information together
- Outperforms other state-of-the-art collaborative filtering approaches

Future Work

- Explore the relationship between user information and item information
- Scalability analysis and improvement of our algorithm
- Employ more metrics to measure our algorithm
Conclusions and Future Work

Q & A

- Home Page: http://www.cse.cuhk.edu.hk/~hma
- Email: hma@cse.cuhk.edu.hk
- Thanks!