Recommendation Systems
◆ Predict items a user may be interested in based on information about the user and the items
◆ An effective way to help people cope with information overload
◆ Examples: Amazon, Netflix, Tivo, ...

So How Can We Do It?
◆ The content based approach
◆ The user feedback based approach

Collaborative Filtering
◆ Rate items based on the ratings of other users who have similar taste as you

Problem Definitions
◆ Prediction
  ■ Given: a user and \( k \) items
  ■ Return: predicted rating for each item
◆ Recommendation
  ■ Given: a user
  ■ Return: \( k \) items from the database with the highest predicted rating

Basic Assumptions
◆ Items are evaluated by users explicitly or implicitly
  ■ Ratings, reviews
  ■ Purchases, browsing behaviors
  ■ ...
◆ We may map explicit and implicit evaluations to a rating scale, e.g. 1-5.
Heuristic
- People who agreed in the past are likely to agree in the future.

Problem Formulation
- User-Item Matrix

<table>
<thead>
<tr>
<th>Item</th>
<th>Ken</th>
<th>Lee</th>
<th>Meg</th>
<th>Nan</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>??</td>
<td>2</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

So what would be Ken's rating for Item 6?

Solving the Problem
- Intuition: Ken's rating for Item 6 is likely to be high
  - Ken's ratings are similar to Meg's
  - Ken's ratings are opposite of Lee's
- Develop the algorithm
  1. Calculate the similarity between users
  2. Combine the ratings by similar users

Similarity Measure
- Pearson Correlation Coefficient
  - A measure of linear correlation of two random variables

Pearson Correlation Coefficient
- Let \( x \) and \( y \) be two users, and \( r_{x,i} \) be the rating of item \( i \) by user \( x \)

\[
 w_{x,y} = \frac{\text{cov}(x, y)}{\sigma_x \sigma_y} = \frac{\sum (r_{x,i} - \bar{r}_x) (r_{y,i} - \bar{r}_y)}{\sqrt{\sum (r_{x,i} - \bar{r}_x)^2} \sqrt{\sum (r_{y,i} - \bar{r}_y)^2}}
\]

So what is \( w_{ken,lee} \)? What's the range of \( w_{ij} \)?

Predict the Rating
- The predicted rating \( P_{x,i} \) should be a function of
  1. The past ratings of user \( x \)
  2. The ratings of other users for item \( i \), weighted by their similarity to user \( x \)
Predicted Rating

- $p_{x,i}$ is the predicted rating of item $i$ by user $x$
- $p_{x,i} = \bar{r}_x + \frac{\sum (r_{u,i} - \bar{r}_x) \times w_{x,u}}{\sum w_{x,u}}$

So what is $p_{x,u}$ ??

Variations and Optimizations

- Similarity measure
- Significance weighting
- Item rating variance
- Neighborhood selection
- Combine neighborhood ratings

Other Similarity Measures ...

- Spearman Correlation
  - Uses ranks instead of raw rating scores
- Cosine similarity
- Mean squared difference
- Entropy-based
- ...

... Other Similarity Measures

Cosine similarity: $\cos(X, Y) = \frac{X \cdot Y}{\|X\| \|Y\|} = \frac{\sum x_i y_i}{\sqrt{\sum x_i^2} \sqrt{\sum y_i^2}}$

Mean squared difference: $\text{msd}(X, Y) = \frac{\sum (x_i - y_i)^2}{N}$

Entropy-based association: $h(X, Y) = -\sum p_{x,i} \ln p_{x,i}$

Significance Weighting

- Weight users in additional to the similarity measure
- $w = \begin{cases} 1 & n \geq 50 \\ n/50 & n < 50 \end{cases}$

where $n$ is the number of items rated by both users.

Item Rating Variance

- Some items are more telling about tastes than others
  - E.g. “Sleepless in Seattle” is more telling about taste than “Titanic”
  - Give more weight to items with high variance in ratings

Some items are more telling about tastes than others
- E.g. “Sleepless in Seattle” is more telling about taste than “Titanic”
- Give more weight to items with high variance in ratings
Neighborhood Selection
- Select a subset of users for better performance and **accuracy**
  - Correlation threshold
  - Best $n$ neighbors

Algorithm Quality Metrics
- **Coverage** – percentage of items for which the system can produce a prediction
- **Accuracy**
  - Statistical metrics
    - Mean Absolute Error (MAE)
  - Decision-support metrics
- **Efficiency**
  - Throughput – number of recommendations per second

Combine Neighborhood Ratings
- Deviation from mean
- Weighted average
- Weighted average of z-scores

$$\text{Mean absolute deviation:} \quad \frac{1}{n} \sum_{i=1}^{n} |r_i - \bar{r}|$$

$$\text{Standardized measurement (z-score):} \quad z_i = \frac{r_i - \bar{r}}{s}$$

And The Winners Are
- **Similarity measure**
  - Pearson Correlation
  - Spearman Correlation
- **Significance weighting**
- **Neighborhood selection**
  - Best $n$ neighbors with $n=20$
- **Combine neighborhood ratings**
  - Deviation from mean

Other Recommendation Algorithms
- Combine collaborative and content-based filtering
- Item-item collaborative filtering
- Bayesian networks
- ...

References