CSE 291:

Trends in Recommender Systems and Human Behavioral Modeling

Week 9 presentations
Personalized Key Frames Recommendation

Presented by Kriti Aggarwal
Problem description

- Diverse interests on the contents even for same video.
- Model time synchronized comments and the video images for finding personalized key frames for different users.
Input

- Visual features:
  - Key frame image features
- Text features:
  - Time synchronized comments

Figure 1: A simple example of TSC. Different users may express real-time opinions directly upon their interested frames. The comments are manually translated into English by the authors.
Dataset: Video sharing website (Bilibili)

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of users (#users)</td>
<td>1,133,750</td>
</tr>
<tr>
<td>Total number of movies (#movies)</td>
<td>7,166</td>
</tr>
<tr>
<td>Total number of TSCs (#TSCs)</td>
<td>11,842,166</td>
</tr>
<tr>
<td>Ave. TSCs per movie</td>
<td>1,652.5</td>
</tr>
<tr>
<td>Ave. users per movie</td>
<td>465.9</td>
</tr>
<tr>
<td>Ave. TSCs per user</td>
<td>10.4</td>
</tr>
<tr>
<td>Max/Min number of TSCs for a movie</td>
<td>8,028/101</td>
</tr>
<tr>
<td>Max/Min number of users for a movie</td>
<td>3,370/1</td>
</tr>
<tr>
<td>Max/Min number of TSCs for a user</td>
<td>68,236/1</td>
</tr>
</tbody>
</table>
Methodology

- Use neural network based collaborative filtering for modelling user and key frame embeddings.
- Use long-short term memory network for modelling the user comments.
- Integrate the two methods to predict the next key frame.
Matrix factorization using neural network

- Find user and item embeddings using neural network.
Image based model

- Preprocessed image features are merged with the frame latent factors to derive new embedding.
- Which is used to generate prediction.
Image based model

\[ q^*_k = \text{MERGE}(q_k, W^{\text{image}} \cdot \nu s l_k) \]

\[ \text{MERGE}((a_1, a_2, \ldots, a_K), (b_1, b_2, \ldots, b_K)) = (a_1 b_1, a_2 b_2, \ldots, a_K b_K) \]

\[ \hat{y}^{\text{image}}_{uk} = \text{LOGISTIC}(p_u \cdot q^*_k). \]
Loss function for image based features

\[ L_1 = \log \prod_{(u,k)} (\hat{y}_{uk}^{\text{image}})^{y_{uk}} (1 - \hat{y}_{uk}^{\text{image}})^{1-y_{uk}} \]

\[ = \log \prod_{(u,k) \in O^+} \hat{y}_{uk}^{\text{image}} \prod_{(u,k) \in O^-} (1 - \hat{y}_{uk}^{\text{image}}) \]

\[ = \sum_{(u,k) \in O^+} \log \hat{y}_{uk}^{\text{image}} + \sum_{(u,k) \in O^-} \log (1 - \hat{y}_{uk}^{\text{image}}) \]
Text based model

- Preference embedding is added to every time step in LSTM.

- LSTM captures the words in the comments.
Text based model

\[
h_1 = LSTM(w_{tscuk}^0, e_{uk}^{pre_0})
\]

\[
h_t = LSTM(h_{t-1}, w_{tscuk}^{t-1}, e_{uk}^{pre_0}) \quad t \in \{2, 3, \ldots s_{uk}\}
\]

\[
p(w_{tscuk}^t | e_{uk}^{pre_0}, w_{tscuk}^{0:t-1}) = \text{SOFTMAX}(h_t)
\]
Loss for Text based model

\[ L_2 = \sum_{(u,k) \in O^+ \cup O^-} \sum_{t=1}^{s_{uk}-1} \log p(w_{tsc_{uk}}^t | e_{uk}^{pre_0}, w_{tsc_{uk}}^{0:t-1}) \]

\[ + \sum_{(u,k) \in O^+} \log \hat{y}_{uk}^{TSC} + \sum_{(u,k) \in O^-} \log (1 - \hat{y}_{uk}^{TSC}) \]
Integrated model: Baseline
Integrated model
Integrated Model

\[
\begin{align*}
    e_{uk}^{pre_0} &= p_u \odot q_k^* \\
    e_{uk}^{pre_i} &= g^{nl}(W^i \cdot e_{uk}^{pre_{i-1}}) \quad i \in \{1, 2, \ldots, D\} \\
    \hat{y}_{uk}^{\text{integrated}} &= \text{LOGISTIC}(w^{output} \cdot e_{uk}^{pre_D}) \\
    L_3 &= \alpha \sum_{(u,k) \in O^+ \cup O^-} \sum_{t=1}^{s_{uk}-1} \log p\left(w_{tsck_k}^t \mid e_{uk}^{pre_{init}}, w_{tsck_k}^{0:t-1}\right) + \\
    &\left(1 - \alpha\right) \left(\sum_{(u,k) \in O^+} \log \hat{y}_{uk}^{\text{integrated}} + \sum_{(u,k) \in O^-} \log \left(1 - \hat{y}_{uk}^{\text{integrated}}\right)\right)
\end{align*}
\]
Evaluation method

- If a user comments on a keyframe with positive sentiment, then this frame would be the one that attracts her, so the empirical experiments are conducted by comparing the predicted keyframes with the true positive ones.

- 30% of each user’s positive keyframes are selected as the test dataset, while the others are used for training.
Results
Key takeaways

- Combines model-based collaborative filtering and long-short term memory network together.
- To model user commented keyframes and time synchronized comments simultaneously.
Questions?
When do Recommender Systems Work the Best?
The Moderating Effects of Product Attributes and
Consumer Reviews on Recommender Performance
Dokyun Lee, Kartik Hosanagar

Presented by Siyu Jiang
Motivation

1. Using Recommender Systems:
   ○ Only 15% of the companies are getting good return on investment.
   ○ 72% attributed failure to lack of knowledge on recommender systems.

2. No literatures on how product attributes and review ratings can affect recommender systems.
This Work

- Cooperated with a top e-commerce site.
- Augmented the dataset with AMT tagged attributes and review ratings.
- Studied their moderating influence on the recommender systems.
Data

- Complete user-item level *views* and *purchase transactional data*.
- Augmented with 1) review data from the website and 2) item attributes.
Data - users, items

355,084 user-item transactional records.

92,188 treated users, 92,187 control users.

37,215 items.
Data - Recommender system

Simple collaborative filtering recommender system - “People who purchased this item also purchased”.

People Who Purchased This Item Also Purchased

1. **DHP Metal Frame Futon - BLACK**
   - Rating: ★★★☆☆
   - Reviews: 35 reviews
   - Price: $164
   - Add to cart

2. **Delaney Split Futon**
   - Rating: ★★★★☆
   - Reviews: 6 reviews
   - Price: $299
   - Add to cart

3. **DHP Delaney Sofa Sleeper**
   - Rating: ★★★☆☆
   - Reviews: 8 reviews
   - Price: $300
   - Add to cart

4. **MAINSTAYS Faux Leather Sofa Bed - Black**
   - Rating: ★★★★☆
   - Reviews: 35 reviews
   - Price: $198
   - Add to cart

5. **Ara Studio Sleeper Futon**
   - Rating: ★★★☆☆
   - Reviews: 2 reviews
   - Price: $499
   - Add to cart

6. **Delaney Split Futon - BROWN**
   - Rating: ★★★☆☆
   - Reviews: 8 reviews
   - Price: $299
   - Add to cart
Product Attributes

- It is infeasible to go through all product attributes.
- Focused on attributes that:
  - are shown in online review literatures to influence consumer purchase.
  - are clear and simple.
  - have strong theoretical background.
Product Attributes - Hedonic VS. Utilitarian

- **Hedonic**: consumption is primarily characterized by an affective and sensory experience of aesthetic or sensual pleasure, fantasy, and fun.
- **Utilitarian**: consumption is cognitively driven, instrumental, goal-oriented, and accomplishes a function or practical task.
Product Attributes - Hedonic VS. Utilitarian

Hypothesis:

1. The base conversion rate for utilitarian goods will be higher.
2. The increase in conversion rate by using recommender will be higher for hedonic goods compared to utilitarian goods.
Product Attributes - Search VS. Experience

1. Search goods: consists of attributes that can easily be discerned before purchase and are dominated by objective attributes, such as computer memory.

2. Experience goods: consists of attributes that cannot be easily discerned before purchase and are dominated by subjective attributes, such as taste of wine.
Product Attributes - Search VS. Experience

Hypothesis:

1. The base conversion rate for search goods will be higher.
2. The increase in conversion rate under the use of a recommender will be higher for experience goods, compared to search goods.
Consumer Reviews (Ratings)

Hypothesis:

1. The base conversion rate will be increased with higher ratings.
2. The positive impact on conversion rate from high ratings will be lessened under the presence of a recommender system.
Other Product Attributes

Price, Brand, Durability, Length of item Description...
Model - linear probability model

Baseline hypotheses:

\[ P(\text{conversion})_{iu} = \beta_0 + \beta_1 \text{PRICE}_i + \beta_2 \text{REC}_u \]
\[ + \beta_3 \text{UTILHEDO}_i + \beta_4 \text{SEARCHEXP}_i \]
\[ + \beta_5 \text{DURABILITY}_i + \beta_6 \text{BRAND}_i + \beta_7 \text{DESLEN}_i \]
\[ + \beta_8 \text{AVGRATINGS}_i + \beta_9 \text{RATINGNUMB}_i \]
\[ + \beta_{10} \text{PRICE}_i \times \text{REC}_u + \beta_{10} \text{UTILHEDO}_i \times \text{REC}_u \]
\[ + \beta_{11} \text{SEARCHEXP}_i \times \text{REC}_u + \beta_{10} \text{DESLEN}_i \times \text{REC}_u \]
\[ + \beta_{10} \text{AVGRATINGS}_i \times \text{REC}_u \]
\[ + \beta_{11} \text{RATINGNUMB}_i \times \text{REC}_u + \epsilon_u \]

Interaction with recommenders:
Results

The result of previous hypothesis can be shown from the coefficients.

Table 5: Main Results Table:

\[ * = p-value < 0.05, ** = p-value < 0.01, *** = p-value < 0.001 \]

<table>
<thead>
<tr>
<th>Variables</th>
<th>Estimate</th>
<th>Std. Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant</td>
<td>0.034771</td>
<td>0.001175</td>
</tr>
<tr>
<td>PRICE</td>
<td>-0.000019</td>
<td>0.00003</td>
</tr>
<tr>
<td>REC</td>
<td>0.002797</td>
<td>0.001042</td>
</tr>
<tr>
<td>DESLEN</td>
<td>-0.000001</td>
<td>0.000001</td>
</tr>
<tr>
<td>AVGRATING</td>
<td>0.002013</td>
<td>0.000153</td>
</tr>
<tr>
<td>RATINGNUMB</td>
<td>-0.000002</td>
<td>0.000003</td>
</tr>
<tr>
<td>UTILHEDO (UTIL=1)</td>
<td>0.005120</td>
<td>0.000677</td>
</tr>
<tr>
<td>SEARCHEXP (SEA=1)</td>
<td>0.003207</td>
<td>0.000677</td>
</tr>
<tr>
<td>BRAND</td>
<td>0.001941</td>
<td>0.000681</td>
</tr>
<tr>
<td>DURABILITY</td>
<td>-0.004763</td>
<td>0.00018</td>
</tr>
<tr>
<td>REC X PRICE</td>
<td>-0.000010</td>
<td>0.000004</td>
</tr>
<tr>
<td>REC X DESLEN</td>
<td>0.000005</td>
<td>0.000002</td>
</tr>
<tr>
<td>REC X AVGRATING</td>
<td>-0.000772</td>
<td>0.000215</td>
</tr>
<tr>
<td>REC X RATINGNUMB</td>
<td>0.000011</td>
<td>0.000004</td>
</tr>
<tr>
<td>REC X UTILHEDO</td>
<td>-0.003064</td>
<td>0.000944</td>
</tr>
<tr>
<td>REC X SEARCHEXP</td>
<td>0.000148</td>
<td>0.000945</td>
</tr>
</tbody>
</table>
Results - Hedonic VS. Utilitarian

Table 5: Main Results Table:

\(\text{\footnotesize \(*\)} = p\text{-value} < 0.05, \text{\footnotesize \(**\)} = p\text{-value} < 0.01, \text{\footnotesize \(***\)} = p\text{-value} < 0.001\)

<table>
<thead>
<tr>
<th>Variables</th>
<th>Estimate</th>
<th>Std Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant</td>
<td>0.034771</td>
<td>0.001175</td>
</tr>
<tr>
<td>PRICE</td>
<td>-0.000019</td>
<td>0.000003</td>
</tr>
<tr>
<td>REC</td>
<td>0.002797</td>
<td>0.001042</td>
</tr>
<tr>
<td>DESLEN</td>
<td>-0.000001</td>
<td>0.000001</td>
</tr>
<tr>
<td>AVGRATING</td>
<td>0.002013</td>
<td>0.000153</td>
</tr>
<tr>
<td>RATINGNUMB</td>
<td>-0.000002</td>
<td>0.000003</td>
</tr>
<tr>
<td>UTILHEDO (UTIL=1)</td>
<td>0.005120</td>
<td>0.000677</td>
</tr>
<tr>
<td>SEARCHEXP (SEA=1)</td>
<td>0.003207</td>
<td>0.000677</td>
</tr>
<tr>
<td>BRAND</td>
<td>0.001941</td>
<td>0.000681</td>
</tr>
<tr>
<td>DURABILITY</td>
<td>-0.004763</td>
<td>0.00018</td>
</tr>
<tr>
<td>REC X PRICE</td>
<td>-0.000010</td>
<td>0.000004</td>
</tr>
<tr>
<td>REC X DESLEN</td>
<td>0.000005</td>
<td>0.000002</td>
</tr>
<tr>
<td>REC X AVGRATING</td>
<td>-0.000772</td>
<td>0.000215</td>
</tr>
<tr>
<td>REC X RATINGNUMB</td>
<td>0.000011</td>
<td>0.000004</td>
</tr>
<tr>
<td>REC X UTILHEDO</td>
<td>-0.003064</td>
<td>0.000944</td>
</tr>
<tr>
<td>REC X SEARCHEXP</td>
<td>0.000148</td>
<td>0.000945</td>
</tr>
</tbody>
</table>
Results - Hedonic VS. Utilitarian

Hypothesis:

1. The base conversion rate for utilitarian goods will be higher. ✓
2. The increase in conversion rate by using recommender will be higher for hedonic goods compared to utilitarian goods. ✓
### Results - Search VS. Experience

<table>
<thead>
<tr>
<th>Variables</th>
<th>Estimate</th>
<th>Std Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant</td>
<td>0.034771</td>
<td>0.001175</td>
</tr>
<tr>
<td>PRICE</td>
<td>-0.000019</td>
<td>0.000003</td>
</tr>
<tr>
<td>REC</td>
<td>0.002797</td>
<td>0.001042</td>
</tr>
<tr>
<td>DESLEN</td>
<td>-0.000001</td>
<td>0.000001</td>
</tr>
<tr>
<td>AVGRATING</td>
<td>0.002013</td>
<td>0.000153</td>
</tr>
<tr>
<td>RATINGNUMB</td>
<td>-0.000002</td>
<td>0.000003</td>
</tr>
<tr>
<td>UTILHEDO (UTIL=1)</td>
<td>0.005120</td>
<td>0.000677</td>
</tr>
<tr>
<td>SEARCHEXP (SEA=1)</td>
<td>0.003207</td>
<td>0.000677</td>
</tr>
<tr>
<td>BRAND</td>
<td>0.001941</td>
<td>0.000681</td>
</tr>
<tr>
<td>DURABILITY</td>
<td>-0.004763</td>
<td>0.00018</td>
</tr>
<tr>
<td>REC X PRICE</td>
<td>-0.000010</td>
<td>0.000004</td>
</tr>
<tr>
<td>REC X DESLEN</td>
<td>0.000005</td>
<td>0.000002</td>
</tr>
<tr>
<td>REC X AVGRATING</td>
<td>-0.000772</td>
<td>0.000215</td>
</tr>
<tr>
<td>REC X RATINGNUMB</td>
<td>0.000011</td>
<td>0.000004</td>
</tr>
<tr>
<td>REC X UTILHEDO</td>
<td>-0.003064</td>
<td>0.000944</td>
</tr>
<tr>
<td>REC X SEARCHEXP</td>
<td>0.000148</td>
<td>0.000945</td>
</tr>
</tbody>
</table>

\( ^* = p-value < 0.05, \quad ^** = p-value < 0.01, \quad ^*** = p-value < 0.001 \)
Results - Search VS. Experience

Hypothesis:
1. The base conversion rate for search goods will be higher. ✓
2. The increase in conversion rate under the use of a recommender will be higher for experience goods, compared to search goods. ×
### Results - Review Ratings

#### Table 5: Main Results Table:

*\( \ast \) = p-value < 0.05, *\( \ast \ast \) = p-value < 0.01, *\( \ast \ast \ast \) = p-value < 0.001

<table>
<thead>
<tr>
<th>Variables</th>
<th>Estimate</th>
<th>Std Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant</td>
<td>0.034771</td>
<td>0.001175</td>
</tr>
<tr>
<td>PRICE</td>
<td>-0.000019</td>
<td>0.000003</td>
</tr>
<tr>
<td>REC</td>
<td>0.002797</td>
<td>0.001042</td>
</tr>
<tr>
<td>DESLEN</td>
<td>0.000001</td>
<td>0.000001</td>
</tr>
<tr>
<td><strong>AVGRATING</strong></td>
<td><strong>0.002013</strong>*</td>
<td><strong>0.000153</strong>*</td>
</tr>
<tr>
<td>RATINGNUMB</td>
<td><strong>-0.000002</strong></td>
<td><strong>0.000003</strong></td>
</tr>
<tr>
<td>UTILHEDO (UTIL=1)</td>
<td>0.005120</td>
<td>0.000677</td>
</tr>
<tr>
<td>SEARCHEXP (SEA=1)</td>
<td>0.003207</td>
<td>0.000677</td>
</tr>
<tr>
<td>BRAND</td>
<td>0.001941</td>
<td>0.000681</td>
</tr>
<tr>
<td>DURABILITY</td>
<td>-0.004763</td>
<td>0.00018</td>
</tr>
<tr>
<td>REC X PRICE</td>
<td>-0.000010</td>
<td>0.000004</td>
</tr>
<tr>
<td>REC X DESLEN</td>
<td>0.000005</td>
<td>0.000002</td>
</tr>
<tr>
<td><strong>REC X AVGRATING</strong></td>
<td><strong>-0.000772</strong>*</td>
<td><strong>0.000215</strong>*</td>
</tr>
<tr>
<td>REC X RATINGNUMB</td>
<td>0.000011</td>
<td>0.000004</td>
</tr>
<tr>
<td>REC X UTILHEDO</td>
<td>-0.003064</td>
<td>0.000944</td>
</tr>
<tr>
<td>REC X SEARCHEXP</td>
<td>0.000148</td>
<td>0.000945</td>
</tr>
</tbody>
</table>
Results - Review Ratings

Hypothesis:

1. The base conversion rate will be increased with higher ratings. ✓
2. The positive impact on conversion rate from high ratings will be lessened under the presence of a recommender system. ✓
Conclusion

● This paper is the first to examine the interaction between a recommender system and product attributes/review ratings by doing experiments with a large e-commerce company.
● Managers could use the results to implement more effective recommender systems.
● Weaknesses:
  ○ Dividing products into two types is not elegant.
  ○ Only used simple collaborative filtering.
Questions?

Do you think this metric is a little strict to measure the effect of recommender systems? For any website other than e-commerce (like youtube etc.), a click would be a strong signal of interest. Based on that, do you think monitoring the click behavior would be a better choice for measuring the impact of recommender systems?
Using Navigation to Improve Recommendations in Real Time

(Wu et al., 2013) - Akanksha Grover
Motivation

- Inferring **within-session user intent on-the-fly** based on navigation interactions
- Using **navigation patterns** and adapting recommendations in real-time
- Carried out entirely in the client (such as a browser) **without added latency**
- User’s intention can vary between sessions in ways that cannot be captured by prior knowledge (ex. Movie with friends and family, dedicated vs background viewing etc.)
- Difficult to predict a user’s intention when recommendations are typically generated — before their session has begun (context aware systems solve only a small portion of this using time of day, geo tags etc.)
Relevance Feedback in Search

- A lot of works have explored user interactions like mouse cursor movements, text selection events in search, eye gaze positions etc. to understand user intent.
- However, all work uses such data to infer a posteriori rather than using it immediately.
- This is even more pertinent in lean back recommendation experiences (e.g. Android TV, Playstation, Apple TV, Roku).
- Some systems use query chains to guide the users towards a better match.
- A new engine-Surf Canyon are able to alter the result page dynamically based on user’s interaction with the page. Based on clicks in previous search queries, the system is able to alter results for future queries in the same session.
Context in Recommender Systems

- Factorization-based linear models are among the most effective and popular recommendation approaches that use context for recommendation.
- But joint models of content consumption and navigation signals are still lacking.
- Provide additional relevant recommendations by reordering rows of items.
- Model characteristics:
  - Probabilistic user model: jointly navigation and consumption through a latent interest variable.
  - Hybrid Inference: uses both offline inference to learn static user preferences as well as online inference through Expectation Maximization.
  - Cold-start Experiments: Experiments that show better results for cold
User Interface

- A page is composed of a list of rows, each of which is composed of a list of videos.

- Videos in a row are topically or contextually coherent to ease user navigation (e.g. "TV Comedies" or "Popular on Netflix"). Users can scroll vertically to see different rows and horizontally to see videos.

- We assume that we have a base recommender system that selects about 40 rows from all possible rows types, and each of which selects 75 personalized candidate videos from our full catalog.

- Goal is to optimize the ranking of the rows and also the ranking of videos within each of them.
Model

- Design a statistical model that addresses following aspects of user interactions:
  - Play Probabilities
  - Per session User Intent Model
  - Full Graphical Model
  - Ranking Procedure
  - User Fatigue behaviour to improve accuracy

- The instant implicit feedback can be used in any model where the entire interface does not fit on a single display (e.g. maps, search results, messages) and the model can incorporate navigation data.
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_i$</td>
<td>$i$-th video of a row</td>
</tr>
<tr>
<td>$r_i$</td>
<td>$i$-th row in a page</td>
</tr>
<tr>
<td>$\rho_r$</td>
<td>Row type of row $r$</td>
</tr>
<tr>
<td>$\psi_*$</td>
<td>Parameters for play probability model</td>
</tr>
<tr>
<td>$v_*$</td>
<td>Factorization-based parameters for interest model</td>
</tr>
<tr>
<td>$w_*$</td>
<td>Feature-based parameters for interest model</td>
</tr>
<tr>
<td>$f_t$</td>
<td>$\in \mathbb{R}_+^d$, vector of $d$ features of video $t$</td>
</tr>
<tr>
<td>$f_{(row)}^r$</td>
<td>$\in \mathbb{R}_+^d$, vector of $d$ features of row $r$</td>
</tr>
<tr>
<td>$S$</td>
<td>$\in {0, 1}$, scroll indicator</td>
</tr>
<tr>
<td>$C$</td>
<td>$\in {0, 1}$, play indicator</td>
</tr>
<tr>
<td>$I$</td>
<td>$\in {0, 1}$, interest indicator</td>
</tr>
</tbody>
</table>
Play Prediction (1/2)

- Use the concept of **submodularity** - added benefit from recommending a movie is not determined in isolation but in terms of how novel it is relative to the previous recommendations.

\[
\mathbb{P}(C_i = 1) = \sigma(g(t_{0:i}, \Psi))
\]

\[
g(t_{0:i}, \Psi) = \langle \psi, f_{t_i} \rangle + \langle \tilde{\psi}, q(t_{0:i}) - q(t_{0:i-1}) \rangle
\]

Here the coordinates of \( q \) are given by

\[
[q(t_{0:i})]_j := h \left( \sum_{k=0}^{i} [f_{t_k}]_j \right)
\]

where \( h \) is a suitably chosen concave function. The vectors \( \psi \) and \( \tilde{\psi} \) denote modular and submodular parameters respectively. Each \( q_j \) is a submodular function of a set of videos that captures the diminishing returns effect of feature \( j \). In-
Play Prediction (2/2)

- Intuitively, we assume users become “tired” of a certain feature if they see it in many videos. Taking this into consideration, maximizing play probability leads to a page that has diversity.

- For the purpose of personalization and to share statistical strength, they adopt a hierarchical decomposition of parameters:

\[ \psi = \psi_0 + \psi_u + \psi_{\rho,u} + \psi_{\rho,u} \]

where the terms denote shared, user-specific, row-specific, and {row, user}-specific latent factors respectively.
User Intent (1/2)

- Users generally have relatively stable tastes, they often have different intents in different sessions
- Assume that in each session a user is only interested in some subset of rows
- \( \mathcal{I}_{s,r} \in \{0,1\} \) - indicates a user's interest in row \( r \) of session \( s \)

\[
P(\mathcal{I}_{s,r} = 1|w,v,v_\rho) = \sigma(g'(r)) \tag{3}
\]

where \( g'(r) = \langle w, f_r \rangle + \langle v, v_\rho \rangle \). \hspace{1cm} \tag{4}

Here \( \langle w, f_r \rangle \) is linear with features \( f_r \) and parameters \( w_r \in \mathbb{R}^n \), and \( \langle v, v_\rho \rangle \) is factorization-based with row-type \( \rho \) based latent factors \( v_\rho \) and \( v \). This captures latent co-consumption patterns that are not captured by features. Again, we de-

\[
w = w_0 + w_u + w_s, \\
v = v_0 + v_u + v_s.
\]
User Intent (2/2)

- Horizontal scrolls - navigation signals (rich source of information about user intent)
  \[ S_{s,r} \in \{0, 1\} \]
  Is used model if row \( r \) was scrolled or not

  \[ P(S_{s,r} = 1|I_{s,r} = 1, \delta_{\rho_r}) = \sigma(\delta_{\rho_r}) \]  

  \[ P(S_{s,r} = 1|I_{s,r} = 0, \delta_{\rho_r}) = 0 \]  

  Note that \( P(S_{s,r} = 1|I_{s,r} = 0, \delta_{\rho_r}) = 0 \) by definition since users won’t click on something they are not interested in (\( I_{s,r} = 0 \)). We use \( \rho_r \) to denote the row type of \( r \) and \( \delta_{\rho_r} \) governs the likelihood of scrolling on each row type.

- Play probability of ith video of row \( r \) is given as:

  \[ P(C_{s,r,i} = 1|I_{s,r} = 1, \Psi) = \sigma(g(t_{0:i}, \Psi)) \]
Graphical Model

1. Sample $v_0 \sim \mathcal{N}(0, \lambda_v I)$ and $w_0 \sim \mathcal{N}(0, \lambda_w I)$.
2. For each row type $\rho$,
   (a) Sample $\delta_\rho \sim \mathcal{N}(0, \lambda_\delta)$.
   (b) Sample $v_\rho \sim \mathcal{N}(0, \lambda_v I)$.
3. For each user $u$,
   (a) Sample $v_u \sim \mathcal{N}(0, \lambda_v I)$, $w_u \sim \mathcal{N}(0, \lambda_w I)$.
   (b) For each session $s$ of $u$,
      i. Sample $v_s \sim \mathcal{N}(0, \lambda_v I)$, $w_s \sim \mathcal{N}(0, \lambda_w I)$.
      ii. For each row $r$ with row type $\rho$:
          A. Sample $I_{s,r}$ via (3).
          B. Sample $S_{s,r}|I_{s,r}$ via (5).
          C. For $i$-th video in $r$ sample $C_{s,r,i}$ via (6).

Figure 3: User interaction model. Shaded circles correspond to observed variables, namely play indicators and scroll indicators. For clarity of illustration we omitted the associated hyperparameters from the diagram. $\Theta$ models interest and $\psi$ models the play probabilities.
Ranking Procedure (Online Page Adaptation)

- The goal is to optimize both positions of rows on a page and videos within a row such that the user finds interesting content with minimal effort.
- Use a greedy procedure to find a set of videos that maximize the gain of play probabilities

\[ t_i = \underset{t \in \mathcal{T}_r}{\text{argmax}} \ g(t_{0:i-1} \cap t, \Psi) \]

- Rank the rows according to the probability of play of the videos of the row

\[ r_j = \underset{r \in \mathcal{R}_0, r \notin r_{0:j-1}}{\text{argmax}} \ \mathbb{P} \left( I_r = 1 | v, w \right) \sum_i \mathbb{P} \left( C_{r,t_i} | I_r = 1 \right) \]

- To provide a consistent experience to users, we fix a row once it is seen and only rearrange the rows that a user has yet to see.
Fatigue and Repeated Plays

- Control for the fact that a user might have seen a recommendation previously or watched the movie before on the service.

\[
\text{fatigue}(x_t) = \begin{cases} 
    a_t x_t + b_t & \text{if } x_t < k. \\
    a_t k + b_t + d_t (x - k) & \text{otherwise}
\end{cases} \quad (8)
\]

Here $a_t, b_t, d_t \in \mathbb{R}$ control slope, offset and secondary slope of the fatigue function. $k \in \mathbb{Z}^+$ determines the position of the slope change, and $x_t$ is the number of previous impressions. As before we can decompose $a_t$ (similarly for $b_t, d_t$) into $a_{u,t} + a_{0,t}$ to prevent overfitting.
Inference (1/3)

- Use negative log-posterior of the data to obtain maximum-a-posteriori estimate

\[ L := - \log \mathbb{P}(S, C|\text{rest}) + \Omega(\Theta, \Psi, \delta), \]

Here the first term maximizes the likelihood while the second term is a regularizer that penalizes complex models.

\[ \log \mathbb{P}(S, C|\text{rest}) = \sum_s \sum_r \log \mathbb{P}(S_{s,r}, C_{s,r,,:}|\text{rest}) \quad (10) \]

\[ \mathbb{P}(S_{s,r}, C_{s,r,:}|\text{rest}) = \sum_{j \in \{0,1\}} \mathbb{P}(I_{s,r} = j|\Theta) \mathbb{P}(S_{s,r}|I_{s,r} = j, \delta) \prod_i \mathbb{P}(C_{s,r,i}|I_{s,r} = j, \Psi) \quad (11) \]
Inference (2/3)

- Offline Training

**E-step** Here we estimate the posterior probability of $I$ with fixed $\{\Phi, \Theta, \delta\}$. We define

$$Q(I_s, r)_j := \mathbb{P}(I_s, r = j| S_s, r, C_s, r, \cdot)$$

$$\propto \prod_i \mathbb{P}(C_s, r, i| I_s, r = j, \Psi) \mathbb{P}(S_s, r| I_s, r = j, \delta)$$

$$\mathbb{P}(I_s, r = j| \Theta)$$  \hspace{1cm} (12)

**M-step** Next we optimize $\Psi, \Theta,$ and $\delta$ with $I_s, r$ weighted by the posterior $Q(I_s, r)_j$. Define

$$J(Q, \Psi, \Theta, \delta) := \sum_{j \in \{0, 1\}} Q(I_s, r)_j \log \frac{\mathbb{P}(I_s, r = j, S_s, r, C_s, r, \cdot)}{Q(I_s, r)_j}$$

$$\{\Psi, \Theta, \delta\} = \arg\min_{\Psi, \Theta, \delta} -J(Q, \Psi, \Theta, \delta) + \lambda \| (\Psi, \Theta, \delta) \|_2^2$$

In experiments we use stochastic gradient descent for offline inference. It randomly picks a session from training set and performs a gradient step with gradient

$$\partial_{\Psi} J = \partial_{\Psi} \sum_{j \in \{0, 1\}} Q(I_s, r)_j \log \mathbb{P}(C_s, r, i| I_s, r = j, \Psi)$$

$$\partial_{\Theta} J = \partial_{\Theta} \sum_{j \in \{0, 1\}} Q(I_s, r)_j \log \mathbb{P}(S_s, r| I_s, r = j, \delta)$$

$$\partial_{\Theta} J = \partial_{\Theta} \sum_{j \in \{0, 1\}} Q(I_s, r)_j \log \mathbb{P}(I_s, r = j| \Theta)$$

Updates with regard to $\Psi$ as videos being weighted by $Q(Is, r)_j$ - This matches intuition: a play/nonplay decision made on videos from a row that a user is interested in should be more important than a decision made on other rows.
Inference (3/3)

- Online Updates
  - An online response and parameter update due to user behavior in the current session.
  - When a user scrolls or skips to the next row without scrolling the MAP estimate of $\Theta_s$ is updated accordingly

$$P(\Theta_s | \text{rest}) \propto \prod_{r \in \text{Seen}} \sum_{j \in \{0,1\}} P(S_{s,r} | I_{s,r} = j, \delta)$$

$$P(I_{s,r} = j | v_s, w_s, \text{rest}) \mathcal{N}(v_s | \lambda_v) \mathcal{N}(w_s | \lambda_w).$$

Here $\text{Seen}$ is the set of displayed rows. We adopt a similar EM strategy. The E-step remains unchanged, while in the M-step we make an update with gradient

$$\partial_{\Theta_s} J(Q, \Psi, \Theta, \delta) = \partial_{\Theta_s} \sum_{j \in \{0,1\}} Q(I_{s,r})_j \log P(I_{s,r} = j | \Theta)$$

Once we update our estimation on $\Theta_s$ and in turn $I_{s,r}$, we can then adapt the page. This is computationally inexpensive because we only need to reorder a small (typically < 40) subset of rows. This is easily feasible by most modern browsers and devices. Most of the terms in (12) and (13) can be precomputed, and computing (and updating) $P(I_{s,r})$ only involves a logarithm and an inner product of sparse feature vectors.
Experiment and Dataset (1/2)

- Real world dataset from Netflix

- Consists of homepage sessions, collected by Netflix for a particular type of playback device from 57,386 distinct profiles pre-computed for members from a single country

- For each video on the page, the dataset also contains an indicator about whether the video was displayed to the user, and whether it was played

- Filter the cases where users are navigating the page to discover new content to watch instead of continuing to watch previously watched shows.

- 294k training and 59k testing sessions.

- Each homepage in the dataset consists of 40 rows that were chosen according to a production personalized recommender system that selects and orders the rows in a personalized fashion for each profile. Thus, for this dataset, the method in this paper already has the space of rows filtered down to a set that are deemed to be relevant
- Evaluate using the following procedure: only use navigation signals up to a certain row (the 10th row in experiments) on a page, and only reorder the rows below that row

- Compare the adapted pages with the pages without adaptation

- Calculate mean reciprocal rank (MRR) and Precision@5. MRR measures how early relevant (played) rows are placed on a page. P@5 measures the fraction of relevant rows in the next 5 rows.

- Compared against ordering the rows on the page via Factorization Machines (FMs) using libFM.

Figure 6: Gains in MRR (left) and Precision@5 (right) as a function of the number of rows visited. We see a clear improvement even just after 2 rows. The Factorization Machine baseline is provided for reference.
Cold Start and Fatigue

Figure 7: Gain in MRR (left) and Precision@5 (right) compared to offline-only model. For users with few training sessions, the benefit gained from online update is the greatest.

Figure 8: Gain in MRR (Top) and Precision@5 (Bottom) obtained by modeling impression fatigue and repeated plays. Performance compared to the model without modeling impression fatigue and repeated plays.
Questions?
An Empirical Analysis of Algorithmic Pricing on Amazon Marketplace

Presented By: Nitin Kalra
Introduction

- Methodology for detection of algorithmic pricing, and use it empirically to analyze their prevalence and behavior on Amazon Marketplace
- Explore the characteristics of the sellers adopting algorithmic pricing and characterize the impact of these strategies on the dynamics of the marketplace
Background

Amazon Marketplace

- Third party sellers
- Fulfilled by Amazon
Buy Box

- Contains the price of the product, shipping information, the name of the seller, and a button to purchase the product.
- 82% of sales go through Buy Box
- Other sellers are listed in “New Offers” page

Figure 1: An example Buy Box on Amazon.
“New Offers” Page

- Ranked listing of Sellers for a given product

Figure 4: An example New Offers page on Amazon, listing all sellers for a given product.
Amazon Marketplace Web Service (MWS)

- A set of APIs for programatically interfacing with the marketplace
- Companies like Sellery, Feedvisor, RepriceIt, etc. leverage MWS to offer subscription-based services for third party sellers that combine inventory management with dynamic pricing capabilities
Data Collection

● Use web scraping to obtain information on the active sellers and their prices.

● Datasets:
  ○ **Crawl1**: Created by frequently crawling pages of 837 best-selling products with at least two sellers. Contains information about all sellers. Does not contain Buy Box information. Crawling done in 2014.
  ○ **Crawl2**: Created by frequently crawling pages of 1000 best selling products. Contains information of up to 20 sellers, as well as Buy Box. Crawling done in 2015.
Analyzing the Buy Box

- Calculate Spearman’s Rank Correlation ($\rho$) between the ordered list of sellers returned by Amazon and the list of sellers sorted by price

- $\rho < 1$ for 20% of products, suggesting Amazon’s systems take additional attributes of Sellers into account while ranking them

*Figure 7: Correlation between price and rank (1 is perfect correlation, -1 is anti-correlation).*
• Relationship between seller rank (from the New Offers page) and the winner of the Buy Box

• Only 60% of the top-ranked sellers win the Buy Box, and there is a long tail of sellers at higher ranks that win.

**Figure 9:** Probability of winning the Buy Box for sellers at different ranks.
Modelling Buy Box as prediction problem

- Random Forest (RF) Classifier to predict whether a specific seller of a product wins the buy box

- Seller features used:
  - Price difference to the lowest
  - Price ratio to the lowest
  - Average rating of seller
  - Positive Feedback percentage
  - Total feedback count
  - Whether product is Fulfilled by Amazon
  - Whether Amazon is the seller.
Results

- Achieves 75–85% accuracy even in the most challenging cases with many sellers
- Baselines achieve only 50–60% accuracy

**Figure 10**: Buy Box winner prediction accuracy for products with different numbers of sellers.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Price Difference to the Lowest</td>
<td>0.36</td>
</tr>
<tr>
<td>Price Ratio to the Lowest</td>
<td>0.33</td>
</tr>
<tr>
<td>Positive Feedback</td>
<td>0.10</td>
</tr>
<tr>
<td>Is Amazon the Seller?</td>
<td>0.10</td>
</tr>
<tr>
<td>Feedback Count</td>
<td>0.06</td>
</tr>
<tr>
<td>Average Rating</td>
<td>0.03</td>
</tr>
<tr>
<td>Is the Product FBA?</td>
<td>0.02</td>
</tr>
</tbody>
</table>

**Table 1:** Relative importance of different features in winning the Buy Box, as determined by our RF classifier.
Dynamic Pricing Detection

- Construct time series of prices for seller/product \((s, r)\) pair:
  \[ S_r = \{ (t_0, p_0), (t_1, p_1), \ldots, (t_m, p_m) \} \]

- Use repricing service “Sellery” to get target prices (lowest price, second lowest price and Amazon price) for a specific product. Construct three target price time series:
  \[
  L O W_r = \{ (t_0, p_0^{low}), (t_1, p_1^{low}), \ldots, (t_m, p_m^{low}) \} \\
  2 N D_r = \{ (t_0, p_0^{2nd}), (t_1, p_1^{2nd}), \ldots, (t_m, p_m^{2nd}) \} \\
  A M Z N_r = \{ (t_0, p_0^{amzn}), (t_1, p_1^{amzn}), \ldots, (t_m, p_m^{amzn}) \}
  \]
- Calculate the similarity between $S_r$ and $LOW_r$, $2ND_r$, and $AMZN_r$ (respectively) using Spearman’s Rank Correlation ($\rho$).
- Mark pairs with $\rho \geq 0.7$ (the empirical cutoff of a strong positive correlation) and p-value $\leq 0.05$ as algorithmic pricing candidates.
- Further filter sellers with at least 20 price changes (change threshold) in the time-series and call them “Algorithmic sellers”.
## Results

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Threshold = 10</th>
<th></th>
<th>Threshold = 20</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sellers</td>
<td>Products</td>
<td>Sellers</td>
<td>Products</td>
</tr>
<tr>
<td>Lowest Price</td>
<td>726</td>
<td>544</td>
<td>426</td>
<td>408</td>
</tr>
<tr>
<td>Amazon Price</td>
<td>297</td>
<td>277</td>
<td>176</td>
<td>183</td>
</tr>
<tr>
<td>2nd Lowest Price</td>
<td>721</td>
<td>494</td>
<td>425</td>
<td>370</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>918</strong></td>
<td><strong>678</strong></td>
<td><strong>543</strong></td>
<td><strong>513</strong></td>
</tr>
</tbody>
</table>

*Table 2:* Number of sellers and products with detected algorithmic pricing, based on two different change thresholds. We use a change threshold of 20 unless otherwise stated.
Figure 14: Example of 3P seller (in red) matching the lowest price of all other sellers.
Figure 16: Example of Amazon (in red) setting a premium over the lowest price of all other sellers.
Figure 17: Example of Amazon (in red) matching to the lowest price over time.
Algorithmic Sellers and Market Dynamics

- Algorithmic sellers are active in the marketplace for significantly longer periods of time than non-algorithmic sellers.
- The median seller/product lifetime for an algorithmic seller is 30 days, while it is only 15 days for a non-algorithmic seller.

**Figure 18:** Distribution of seller/product lifetimes for algorithmic and non-algorithmic sellers.
Algorithmic sellers sell fewer unique products by a large margin.

**Figure 19:** Number of products sold by algorithmic and non-algorithmic sellers.
Algorithmic sellers have slightly higher positive feedback than non-algorithmic sellers.

**Figure 20:** Percentage of positive feedback for algorithmic and non-algorithmic sellers.
Algorithmic sellers acquire significantly greater amounts of feedback, suggesting higher sales volume.

**Figure 21:** Amount of feedback received for algorithmic and non-algorithmic sellers.
The rank of algorithmic sellers on the New Offers page tends to be significantly higher than that of non-algorithmic sellers.

Figure 22: Cumulative distribution of rank on the New Offers page for algo and non-algo sellers.
For non-algorithmic sellers, the price never changes for 65% of the seller/product pairs.

Figure 24: Number of price changes per seller/product pair.
- Products with algorithmic sellers experience many more price and seller changes in the Buy Box.
- 20% of products without algorithmic sellers have zero price changes, versus only 2% for products with algorithmic sellers.

**Figure 25:** Number of changes in the Buy Box for products with and without algorithmic sellers.
In almost all cases, Algorithmic sellers are more likely to win the Buy Box, resulting in higher sales volume.

**Figure 26**: Probability of winning the Buy Box for algo and non-algo sellers at different ranks.
Conclusion

- Algorithmic sellers can be detected using a target price time series, and over 500 such sellers were identified in the data set.

- Algorithmic sellers receive more feedback and win the Buy Box more frequently, likely suggesting higher sales volumes and thus more revenue than non-algorithmic sellers.
Questions?
Modeling the Assimilation-Contrast Effects in Online Product Rating Systems Debiasing and Recommendations

Presented by: Zeng Fan
Overview

- Customer’s rating is distorted by historical ratings
- Customer’s rating behavior can be explained by “Assimilate-Contrast” theory
- Authors propose the *Historical Influence Aware Latent Factor Model* (HIALF) to capture and mitigate historical distortions in each single rating
Dataset

<table>
<thead>
<tr>
<th>category</th>
<th># products</th>
<th># users</th>
<th># ratings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon-books</td>
<td>2,370,585</td>
<td>8,026,324</td>
<td>22,507,155</td>
</tr>
<tr>
<td>Amazon-clothes</td>
<td>1,503,384</td>
<td>3,117,268</td>
<td>5,748,920</td>
</tr>
<tr>
<td>Amazon-electronics</td>
<td>498,196</td>
<td>4,261,096</td>
<td>7,824,482</td>
</tr>
<tr>
<td>Amazon-movies</td>
<td>208,321</td>
<td>2,088,620</td>
<td>4,607,047</td>
</tr>
<tr>
<td>TripAdvisor</td>
<td>12,730</td>
<td>781,329</td>
<td>1,621,956</td>
</tr>
</tbody>
</table>

Processing: group products with similar average ratings into one group such that each group has a maximum deviation of 0.2 in the average rating.

E.g. selected groups of products with average ratings in [2.9, 3.1] has an approximately true quality of 3.
Empirical Measurements

$r_{p,i}$: the $i$th rating of product $p$

$H_{p,i} = (r_{p,1},...,r_{p,i-1})$: $i-1$ ratings of product $p$ received before $r_{p,i}$

$e_{p,i} = \frac{1}{i-1} \Sigma_{k=1}^{i-1} r_{p,k}$: prior expectation formed on historical ratings $H_{p,i}$

$\bar{r}$: average rating

List $\{(e,\bar{r})\}$: describes how prior expectation $e$ affects its next average rating $\bar{r}$
Observations

1. Historical ratings do affect the next rating
2. Each curve is divided into 2 parts by the approximately true quality
“Assimilate-Contrast” Theory

If the disparity between his prior expectation and the product quality can be accepted by the user, the user’s satisfaction with the product assimilates to his prior expectation; otherwise, the difference between the prior expectation and the product quality tends to be magnified.

Examples:

Assimilate: If the user’s experienced quality is 3.5, historical rating is 4, the user is more likely to rate 4.

Contrast: If the user’s experienced quality is 3, historical rating is 4.5, the user is more likely to rate 2

“Assimilate-Contrast” Theory statistically explain how customers are affected by historical rating (see figure (b))
HIALF Model

Recall latent factor model: $r_{u,p} = g + b_u + b_p + x_u^T y_p$

HIALF combines 2 factors together:

1. $q_{u,p}$: User u’s experienced quality of product p

   $$q_{u,p} = g + b_p + x_u^T y_p$$

2. $h_{p,i}$: the distortion from historical ratings

   $$h_{p,i} = f(|H_{p,i}|) \beta(e_{p,i} - q_{u,p})$$

Where categorical function $\beta(x)$ to represent the induced bias when the difference between $e_{p,i}$ and $q_{u,p}$ is $x$,

And $f(x)$ be a scaling function to represent the magnitude of impact by historical ratings of size $x$
HIALF Model

The overall predictor is

\[ \hat{r}_{p,i,u} = b_u + q_{u,p} + \alpha_u h_{p,i} \]

\[ = g + b_u + b_p + x_u^T y_p + \alpha_u f(H_{p,i}) \beta(e_{p,i} - q_{u,p}) \]

Where \( g, b_u, b_p, x_u, y_p \) act the same as latent factor model, \( \alpha_u \) models how the user \( u \) is affected by historical rating
Model $\beta(x)$, $f(x)$, and $e_{p,i}$

- $\beta(x)$ is modeled by non-parametric kernel regression

\[
\beta(x) = \frac{\sum_{l=1}^{n} w(x, e_l) \cdot v_l}{\sum_{l=1}^{n} w(x, e_l)}
\]

- $w(x, x_i)$ gives greater weight to $x$ when it is closer to $x_i$
- $\{e_1, \ldots, e_n\} = \{-4, -3.5, \ldots, 3.5, 4\}$, since both $e_{p,i}$ and $q_{u,p}$ are in $[1,5]$

- $\{v_1, \ldots, v_n\}$ are parameters learned from data
Model $\beta(x)$, $f(x)$, and $e_{p,i}$

- $f(x)$ is modeled so that more historical ratings result in a larger magnifying effect.

$$f(x) = \frac{a}{1 + \exp(-b \cdot x)} - \frac{a}{2}$$

- $e_{p,i}$ is modeled so that users focus more on recent ratings instead of all ratings of a product.

$$e_{p,i} = \frac{\sum_{k=1}^{i-1} \xi(i-k) \cdot r_{p,k}}{\sum_{k=1}^{i-1} \xi(i-k)}$$

where $\xi(d) = \exp(-\gamma \cdot d)$ models the decay of influence.
Model Inference

Solve the following function

\[
\min_{\Theta} \sum_{(p,i,u) \in K} (r_{p,i,u} - \hat{r}_{p,i,u})^2 + \lambda_{\text{rec}}(b_u^2 + b_p^2 + \|x_u\|_2^2 + \|y_p\|_2^2) \\
+ \lambda_f(a^2 + b^2) + \lambda_\beta(\sum_l v_l^2) + \lambda_\alpha(x_u^2)
\]
Experiments
Validating $\beta(x)$ curve

$\beta(x)$ curve matches the “Assimilate-Contrast” theory

Figure 4: The learned disconfirmation bias curve $\beta(x)$. All $\beta(x)$ perfectly match the “Assimilate-Contrast” theory.
Predicting Subsequent Ratings

### Table 3: $MSE$ on five datasets

<table>
<thead>
<tr>
<th></th>
<th>Amazon-movie</th>
<th>Amazon-books</th>
<th>Amazon-electronics</th>
<th>Amazon-clothes</th>
<th>Tripadvisor</th>
</tr>
</thead>
<tbody>
<tr>
<td>HEARD</td>
<td>1.5826</td>
<td>1.5548</td>
<td>3.1170</td>
<td>2.1550</td>
<td>1.3135</td>
</tr>
<tr>
<td>LF</td>
<td>1.2794</td>
<td>1.0777</td>
<td>1.9634</td>
<td>1.4123</td>
<td>1.0074</td>
</tr>
<tr>
<td>HIALF-AVG</td>
<td>1.2054</td>
<td>1.0619</td>
<td>1.9357</td>
<td>1.3985</td>
<td>0.9805</td>
</tr>
<tr>
<td>HIALF</td>
<td><strong>1.1194</strong></td>
<td><strong>1.0318</strong></td>
<td><strong>1.8764</strong></td>
<td><strong>1.3759</strong></td>
<td><strong>0.9405</strong></td>
</tr>
</tbody>
</table>

Benefits of HIALF over HEARD:
- 29.27%
- 32.83%
- 39.80%
- 35.17%
- 28.40%

Benefits of HIALF over LF:
- 12.51%
- 4.26%
- 4.43%
- 2.58%
- 6.64%

HIALF achieves best result. HIALF is more accurate than HIALF-AVG because users is affected more by recent ratings.
Fitting Empirical Observations

HIALF provides the best fit to previous observations in real ratings.
Applications

- Debiased recommender system

\[ \text{rec}(p,u) = g + b_p + b_u + x_u^T y_p \]

- Exposing the intrinsic product quality

\[ q_p^* = \sum_i (g + b_p + x_{\tilde{u}(p,i)}^T y_p) \]
Questions?
Existing works’ deficiencies

HEARD mainly focuses on the macro-level historical ratings’ influence in overall rating distribution.

HIALF focuses on micro-level
Application (cont’d)

Table 4: RMSE on five datasets

<table>
<thead>
<tr>
<th>category</th>
<th>LF</th>
<th>debiased recsys</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon-movie</td>
<td>1.0639</td>
<td>1.0465</td>
</tr>
<tr>
<td>Amazon-books</td>
<td>0.9125</td>
<td>0.8922</td>
</tr>
<tr>
<td>Amazon-electronics</td>
<td>1.2273</td>
<td>1.2083</td>
</tr>
<tr>
<td>Amazon-clothes</td>
<td>1.1239</td>
<td>1.1034</td>
</tr>
<tr>
<td>Tripadvisor</td>
<td>1.1919</td>
<td>1.1776</td>
</tr>
</tbody>
</table>

(a) sample product 1
(b) sample product 2

Figure 7: Two products with similar intrinsic quality have different rating growth histories, leading to significantly distinct ratings.
Leading the Herd Astray

An Experimental Study of Self-Fulfilling Prophecies in an Artificial Cultural Market

Presented by Siddharth Dinesh
27th November, 2017
Cultural Markets and Consumer Decisions

- Cultural Market - Market for books, music, movies etc.
- Information about cultural products available from friends, bestseller lists, box office receipts, online forums
- Consumers’ decisions about cultural products can be influenced by such information

This research was conducted by Salganik and Watts in 2005 and published in 2008.
Theories of Popularity and Self-fulfilling prophecies

Early success leads to future success when information about previous decisions of other customers is made available. Raises the question of whether perceived success alone is sufficient to generate continued success. Is it a self-fulfilling prophecy?

Evidence has been collected in three different levels so far:

1. Individual Level - Placebo effect
2. Dyad Level - Effects of teacher expectations on student outcomes
3. Collective Level - Performativity of economic models, financial panics, investment bubbles

Skeptics contest that observational data alone is insufficient to prove that any real world event was caused by self-fulfillment of false beliefs. The data at the collective level might contain individual level discrepancies in perceived quality which makes it harder to attribute consequences at the collective level to beliefs at a collective level.
Research Questions

Do cultural markets allow for self-fulfilling prophecies?

How much does initial information affect the subsequent success of an item in the market?
Experimental Setup

- Web based experiment
- 12,207 participants could listen to and download songs
- The experiment had 48 relatively unknown songs
- Experiment ran for 21 weeks
Experimental Setup II

- Initial setup period: Two worlds until steady state
- After steady state: 4 worlds

Worlds:

1. Independent
   a. No external information

2. Social Influence
   a. External info about downloads

3. Reversed Social Influence
   a. External info about downloads
   b. Download count is inverted

Before inversion

Social influence

<table>
<thead>
<tr>
<th>Unchanged social influence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inverted social influence #1</td>
</tr>
<tr>
<td>Inverted social influence #2</td>
</tr>
</tbody>
</table>

After inversion

<table>
<thead>
<tr>
<th>Independent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Independent</td>
</tr>
</tbody>
</table>
Results

Result 1:

- Participants who were made aware of the behavior of others were more likely to listen to songs that they believed were more popular
- Slight reversal of pattern for least popular songs

Probability of listening to a song as a function of its popularity in the four worlds
Results II

- In the unchanged world, there is no real change in the trajectory of the downloads of songs
- In the reversed worlds, least popular song earned downloads at faster rate compared to before inversion while more popular songs suffered
- For all songs, initially false perception of popularity caused their real popularity to change in the direction of the false belief

(A) Popularity dynamics before and after the inversion for the most popular song during the set-up period and the least popular
(B) Popularity dynamics for the second most popular song during the set-up period and second least popular
Results III

- Song appeal: $a_i = d/\Sigma d$ in the independent world
- Final ranks are predicted steady state ranks. Calculated by extrapolating trends shown in previous slide
- The “worst” songs were helped by the inversion, the “good” songs were hurt by it, and the success of the “best” songs was largely unaffected

![Relationship between estimated impact of inversion, $\Delta$, and song appeal.](image)
Discussion

- There was a substantial reduction in the overall number of downloads between the unchanged social influence world and the reversed social influence worlds. (2898 vs 2197, 2160)
- Choice to manipulate market information resembles a social dilemma: A common-pool resources dilemma
  - Any individual band could benefit by artificially inflating their perceived popularity, regardless of true popularity or strategy of other bands
  - So it is the rational strategy for all bands to manipulate information
  - However, if all bands use such a strategy, there is a distortion of correlation between appeal and apparent popularity
  - Leads to contraction of the market as a whole, causing all bands to suffer collectively.
Limitations

1. Distortion in the real world occurs repeatedly and is more subtle, has more variety
2. Information in real cultural markets can come from a wide variety of sources. Experiment assumes that all users only used information from website
3. The experiment only had 48 songs
   a. Average participant listened to 1/7th of the music in the market
   b. Some participants listened to almost all the music
   c. This is not feasible in a real cultural market
4. Focus of experiment is solely on consumers
   a. Did not account for decisions of radio stations, cultural critics and music executives

Strengths

1. One of the first large scale web-based experiments in sociological research
2. Use of multiple control groups (worlds) to factor out effect of individual choices from social decisions
Questions?
Post Processing Recommender Systems for Diversity

Presented By: Wen Liang
Motivation

To increase diversity in recommender system

A focus on accuracy or relevance alone can hurt user experience

Popularity bias affects pairing between users and items.

Business need diverse recommendations

User’s intrinsic appreciation for novelty

Smaller display and smaller space (on phones or mobile devices) requires diversity in only a few recommendation.
Graph model, b-matching problem

Model the user-item recommendations provided by CF system as a bipartite

problem, an underlying bipartite graph $G = (L, R)$ with edge

set $E$ is given, along with a nonnegative weight $g$ on the edges,

and two vectors of non-negative integers $(c_1, \ldots, c_l)$ and

$(a_1, \ldots, a_r)$ (degree bounds) such that $\sum_{i=1}^{l} c_i = \sum_{i=1}^{r} a_i$. The goal is to find a maximum $g$-weight (or minimum $g$-cost) subgraph $H$ where the degree of vertex $u_i \in L$ in $H$ is $c_i$ for

every $1 \leq i \leq l$ and the degree of vertex $v_j \in R$ in $H$ is $a_j$ for

every $1 \leq j \leq r$. This problem generalizes the well-known
Discrepancy

The objective for a feasible solution for a subgraph:

\[
D(H) = \sum_{v_j \in R} |deg_H(v_j) - a_j|
\]

The $a_j$ here are the target degree distribution and the discrepancy sums the violations of the degree distribution constraints.

**Min Discrepancy Problem**: minimize the $D(H)$ for a subgraph $H$ with target vector $a$ and vector $c$. 
Post Processing for recommender system

Generate a result with hundreds of recommendation candidates for each user use CF or other algorithm.

These recommendations can be converted to edges in a graph, and construct a bipartite graph with 2 group of users and items.

Use the algorithm to select a subgraph with constraints of target degree (number of recommendations).

The results have higher diversity than just take topK relevant recommendations but cost some precision.
Minimum Cost Flow Problem

- Directed graph \( G = (V, E) \)
- non-negative edge capacities \( u \)
- edge costs \( c \)
- Supply/demand \( b \) on each vertex

\[
\begin{align*}
\text{min} & \quad \sum_{(v,w)\in E} c(v, w)f(v, w) \\
\text{subject to} & \quad f(v, w) \leq u(v, w) \quad \forall (v, w) \in E \\
& \quad \sum_{w \in V} f(v, w) - \sum_{w \in V} f(w, v) = b(v) \quad \forall v \in V \\
& \quad f(v, w) \geq 0 \quad \forall (v, w) \in E
\end{align*}
\]
Algorithm

Each node $u_i$ has supply $c_i$ (specific out-degree).

2 special sink nodes, one with demand $\sum_{j=1}^r a_j$

Assumption to ensure that total supply meets total demand

$$\sum_{i=1}^l c_i = \sum_{i=1}^r a_i$$

Figure 1: The network flow model for the MIN-DISCREPANCY problem with nodes labelled with their supply and arcs labeled with their cost/capacity. Unlabelled nodes have zero supply.
Algorithm

The min-cost flow problem has the same cost as the value of the min-discrepancy problem.

$$\sum_{v_j \in R} |\deg_H(v_j) - a_j| = \sum_{v_j \in P} (\deg_H(v_j) - a_j) + \sum_{v_j \in N} (a_j - \deg_H(v_j))$$

$$\sum_{v_j \in R} (\deg_H(v_j) - a_j) = \sum_{v_j \in R} \deg_H(v_j) - \sum_{i=1}^l c_i = 0$$

The cost from $P$ and $N$ are equal. $$\sum_{v_j \in R} |\deg_H(v_j) - a_j| = 2 \sum_{v_j \in P} (\deg_H(v_j) - a_j)$$

Can be solved in polynomial time $O(|E||V|^2 \log(|V|))$
Two-pass method and weighted method

Two-pass method: first step is to solve the min-discrepancy problem and finds the lowest discrepancy value achievable with given graph for the current target degrees. Second pass selects highest rating solution which achieve this minimum.

Weighted method: Combine these 2 criteria together and the objective is $\text{discrepancy}(H) - \mu \cdot \text{rel}(H)$ with a weight to balance the discrepancy of the subgraph and the average relevance of the recommendations in H as predicted by CF or other recommender.
Greedy Algorithm and Aggregate Diversity

Greedy Algorithm: constructs the solution subgraph iteratively. The choice of edge (recommendation) is conditioned on the discrepancy reduction and quality.

Aggregate Diversity (optimization): set target degree of each item to 1 and solve the min-discrepancy problem.
Experiment

Datasets: MovieLens, Netflix

CF recommender: RankSys project

Generated 240 super-graphs of candidate recommendations

Network flow problem solver: MCFSimplex solver
Evaluation Metrics

For Diversity

**Aggregate Diversity:** The total number of objects that have been recommended to at least one user

**Gini Index:** quantization of wealth and income inequality. (between 0 and 1, lower value means more equitable distribution)

**Entropy:** measure the amount of information contained in a stochastic process.

Discrepancy

For recommendation quality

**Precision:** the fraction of items in the recommendation list which are part of the test set
Approaches and notations

Top: undiversified solution directly from the recommender system

AGG: Aggregate Diversity
GRD: Greedy method
PC: PC Reranking

D@n: Discrepancy
G@n: Gini index

GOL: Two Pass Method
FD: FD Reranking
AB: Bayes Rule Reranking

A@n: Aggregate diversity
E@n: entropy
P@n: Precision

- The aggregate diversity optimization does not yield results that are diverse by the other measures
- Using target distributions that move towards the underlying degree distribution is more effective than using uniform distribution
- Two pass method is very efficient to reduce the normalized discrepancy by 50% with about 15-30% cost of precision. Smooth trade-off between recommendation quality and diversity
Trade-off Between Discrepancy and Precision

Increase the number of candidate recommendations for each user from 100 to 500 in increments of 100.

The fall-off in discrepancy happens first quickly, then slow as more edges are included. More edges, more choices.

Figure 3: Precision discrepancy trade-off in the MovieLens-1m dataset using the MF recommender. In each series, the number of edges in the input graph increases towards the left.
Runtime

Two pass and weighted method do not run as efficiently as reranking methods but provide much better diversification.

Figure 4: Time to optimize the top-10 recommendation task in MovieLens-1m based graphs in seconds ($|L|=5800, |R|=3600$)
Conclusions

- Proposed a new way of measuring the recommendation distribution called discrepancy
- Showed that it can be optimized in polynomial time
- Provided 4 efficient methods
Main Take Aways

- The model only depends on accuracy not performs well in real use
- Some metrics for recommendation diversity evaluation: Discrepancy, aggregate diversity, Gini Index, Entropy
- Min-discrepancy problem can be converted as min-cost problem and be solved in polynomial time
- The methods are effective to improve recommendation diversity
Questions?
Personalized Key Frames Recommendation

Presented by Kriti Aggarwal
Problem description

- Diverse interests on the contents even for same video.
- Model time synchronized comments and the video images for finding personalized key frames for different users.
Input

- Visual features:
  - Key frame image features
- Text features:
  - Time synchronized comments

Figure 1: A simple example of TSC. Different users may express real-time opinions directly upon their interested frames. The comments are manually translated into English by the authors.
Dataset: Video sharing website (Bilibili)

<table>
<thead>
<tr>
<th></th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of users (#users)</td>
<td>1,133,750</td>
</tr>
<tr>
<td>Total number of movies (#movies)</td>
<td>7,166</td>
</tr>
<tr>
<td>Total number of TSCs (#TSCs)</td>
<td>11,842,166</td>
</tr>
<tr>
<td>Ave. TSCs per movie</td>
<td>1,652.5</td>
</tr>
<tr>
<td>Ave. users per movie</td>
<td>465.9</td>
</tr>
<tr>
<td>Ave. TSCs per user</td>
<td>10.4</td>
</tr>
<tr>
<td>Max/Min number of TSCs for a movie</td>
<td>8,028/101</td>
</tr>
<tr>
<td>Max/Min number of users for a movie</td>
<td>3,370/1</td>
</tr>
<tr>
<td>Max/Min number of TSCs for a user</td>
<td>68,236/1</td>
</tr>
</tbody>
</table>
Methodology

- Use neural network based collaborative filtering for modelling user and key frame embeddings.
- Use long-short term memory network for modelling the user comments.
- Integrate the two methods to predict the next key frame.
Matrix factorization using neural network

- Find user and item embeddings using neural network.
Image based model

- Preprocessed image features are merged with the frame latent factors to derive new embedding.
- Which is used to generate prediction.
Image based model

\[ q^*_k = \text{MERGE}(q_k, W^{image} \cdot \nu_{sl_k}) \]

\[ \text{MERGE}((a_1, a_2, \ldots, a_K), (b_1, b_2, \ldots, b_K)) = (a_1 b_1, a_2 b_2, \ldots, a_K b_K) \]

\[ \hat{y}^{image}_{uk} = \text{LOGISTIC}(p_u \cdot q^*_k). \]
Loss function for image based features

\[
L_1 = \log \prod_{(u,k)} (\hat{y}_{uk}^{\text{image}})^{y_{uk}} (1 - \hat{y}_{uk}^{\text{image}})^{1-y_{uk}} \\
= \log \prod_{(u,k) \in O^+} \hat{y}_{uk}^{\text{image}} \prod_{(u,k) \in O^-} (1 - \hat{y}_{uk}^{\text{image}}) \\
= \sum_{(u,k) \in O^+} \log \hat{y}_{uk}^{\text{image}} + \sum_{(u,k) \in O^-} \log (1 - \hat{y}_{uk}^{\text{image}})
\]
Text based model

- Preference embedding is added to every time step in LSTM.

- LSTM captures the words in the comments.
Text based model

\[ h_1 = LSTM(w_{tscuk}^0, e_{uk}^{pre0}) \]

\[ h_t = LSTM(h_{t-1}, w_{tscuk}^{t-1}, e_{uk}^{pre0}) \quad t \in \{2, 3, ..., s_{uk}\} \]

\[ p(w_{tscuk}^t | e_{uk}^{pre0}, w_{tscuk}^{0:t-1}) = \text{SOFTMAX}(h_t) \]
Loss for Text based model

\[
L_2 = \sum_{(u,k) \in O^+ \cup O^-} \sum_{t=1}^{s_{uk}-1} \log p(w_{tscuk}^t | e_{uk}^{pre0}, w_{tscuk}^{0:t-1}) \\
+ \sum_{(u,k) \in O^+} \log \hat{y}_{uk}^{TSC} + \sum_{(u,k) \in O^-} \log (1 - \hat{y}_{uk}^{TSC})
\]
Integrated model: Baseline
Integrated model
Integrated Model

\[ e_{uk}^{pre_0} = p_u \odot q_k^* \]
\[ e_{uk}^{pre_i} = g^{nl}(W^i \cdot e_{uk}^{pre_{i-1}}) \quad i \in \{1, 2, \ldots D\} \]
\[ \hat{y}_{uk}^{\text{integrated}} = \text{LOGISTIC}(w_{\text{output}} \cdot e_{uk}^{pre_D}) \]

\[ L_3 = \alpha \sum_{(u,k) \in O^+ \cup O^-} \sum_{t=1}^{s_{uk}-1} \log p\left(w_{tsc_{uk}}^t | e_{uk}^{pre_{init}}, w_{tsc_{uk}}^{0:t-1}\right) + \]
\[ (1 - \alpha) \left( \sum_{(u,k) \in O^+} \log \hat{y}_{uk}^{\text{integrated}} + \sum_{(u,k) \in O^-} \log \left(1 - \hat{y}_{uk}^{\text{integrated}}\right) \right) \]
Evaluation method

- If a user comments on a keyframe with positive sentiment, then this frame would be the one that attracts her, so the empirical experiments are conducted by comparing the predicted keyframes with the true positive ones.

- 30% of each user’s positive keyframes are selected as the test dataset, while the others are used for training.
Results
Key takeaways

- Combines model-based collaborative filtering and long-short term memory network together.
- To model user commented keyframes and time synchronized comments simultaneously.
Questions?
Exploring the Filter Bubble

The Effect of Using Recommender Systems on Content Diversity


Presented By: Dhruv Sharma
Filter Bubble?

- Coined by Eli Pariser in 2010
- State of intellectual isolation resulted from personalized algorithmic filters
- Pariser claimed that it violates the purpose of internet, i.e, to connect the world
- Talks about filters such as importance, challenging etc, not just relevance
- Ted Talk: [https://www.ted.com/talks/eli_pariser_beware_online_filter_bubbles](https://www.ted.com/talks/eli_pariser_beware_online_filter_bubbles)
Overview of the paper

❖ Talks about filter bubble in recommendation systems from the point of view of content diversity and user response

❖ Key questions:
  ➢ Do recommender systems expose users to narrower content over time?
  ➢ How does the experience of users who take recommendations differ from that of users who do not regularly take recommendations?
Experiments

❖ Dataset: MovieLens (Feb 2008 - Aug 2010)
❖ 2 broad experiments:
  ➢ Analyzing the content diversity for user groups based on consumption of recommendation across time periods
  ➢ Analyzing the user experience based on above groups across time periods
❖ Uses “Top Picks For You” for recommendation set
❖ Key Issue: Unlike NetFlix MovieLens does not have the true data about views and relies on user’s loyalty
- Divide rating history into discrete intervals of 10 ratings (median of the distribution of numbers of ratings per 3 months)
- Reaching a steady state of recommendation:
  - Remove first 3 months + 15 ratings (users had watched many movies before joining MovieLens)
- Selected only users which started using MovieLens between the Feb 2008-Aug 2010
Key Terms

*Following Group*: users who took recommendations in at least 50% of their rating blocks

*Ignoring Group*: users who did not take any recommendations in any of their rating blocks
Key Terms

**Tag Genome:**

- An information space containing a set $M$ of movies, and a set $T$ of tags, where each cell represents relevance of a tag for a movie.
- Users provide the values by evaluating how strongly a tag describe a movie.
Key Terms

**User Experience:**
- The mean in a rating block
- In another definition: positive experience index is defined as the percentile of per use rating average

**Content Diversity:**
- Average pairwise euclidean distances of the movie vectors (represented in the tag genome) (Ziegler et al.) in a set
- Uses euclidean distance as the tag genome is a dense matrix.
- Claimed to be better than computing movie content diversity based on meta-data such as genres, actors, or directors, etc.,
Results - Diversity over time

Do recommender systems expose users to narrower content over time?

<table>
<thead>
<tr>
<th></th>
<th>At the beginning</th>
<th>At the end</th>
<th>Withingroup p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>All users</td>
<td>25.02</td>
<td>24.67</td>
<td>2.43e-06</td>
</tr>
<tr>
<td>Following Group</td>
<td>25.22</td>
<td>24.80</td>
<td>0.014</td>
</tr>
<tr>
<td>Ignoring Group</td>
<td>24.74</td>
<td>24.51</td>
<td>0.087</td>
</tr>
<tr>
<td>Between-group p-value</td>
<td>0.0037</td>
<td>0.0406</td>
<td></td>
</tr>
</tbody>
</table>

Average content diversity of the recommended movies

- The drop in average content diversity for both groups is statistically significant.
- The drop is steeper for *Following Group*, and reduces the gap between the content diversity of the 2 groups.
- Interestingly, the *Ignoring Group* did not take recommendations, still witnessed a drop in content diversity of recommended movies.
Results - User Experience

Does taking recommendations lower the consumed content diversity?

The difference in average content diversity is not significant in the First but becomes significant after using MovieLens.

The reduction in content diversity of consumed content is more for Ignoring Group, thus, Following Group watched more diverse movies.
Results - User Experience

Did the Following Group have better experience?

<table>
<thead>
<tr>
<th>Rating Block</th>
<th>The First</th>
<th>The last</th>
<th>Within-group p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>All users</td>
<td>3.69</td>
<td>3.57</td>
<td>2.2e-16</td>
</tr>
<tr>
<td>Following Group</td>
<td>3.69</td>
<td>3.68</td>
<td>0.7</td>
</tr>
<tr>
<td>Ignoring Group</td>
<td>3.74</td>
<td>3.55</td>
<td>3.128e-11</td>
</tr>
<tr>
<td>Between-group p-value</td>
<td>0.2129</td>
<td>0.001719</td>
<td></td>
</tr>
</tbody>
</table>

Average Rating of the 2 groups

- The difference in average rating was not significant at First, but became significant after using MovieLens
- The *Ignoring* Group saw significant decrease in average rating
- Hence, *Following* Group watched had better experience
Observations

- Taking recommendations reduced the risk of filter bubble
- Following Group saw more diversity in recommendations and consumed content
- Following Group users showed greater satisfaction in terms of mean rating
- The diversity, as well as mean rating, reduced over time
Strengths & Weaknesses

Strengths:
- The paper presents a novel experiment to check the risk of a filter bubble in a recommendation system.

Weaknesses:
- Relies heavily on the fact that users rate each movie they watched anywhere on MovieLens.
- User preferences can be affected by other recommendations.
- Difficult to say if the results can be generalized for other personalized recommendations.
Questions?
Personalized Key Frames Recommendation

Presented by Kriti Aggarwal
Problem description

- Diverse interests on the contents even for same video.
- Model time synchronized comments and the video images for finding personalized key frames for different users.
Input

- Visual features:
  - Key frame image features
- Text features:
  - Time synchronized comments

Figure 1: A simple example of TSC. Different users may express real-time opinions directly upon their interested frames. The comments are manually translated into English by the authors.
Dataset: Video sharing website (Bilibili)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of users (#users)</td>
<td>1,133,750</td>
</tr>
<tr>
<td>Total number of movies (#movies)</td>
<td>7,166</td>
</tr>
<tr>
<td>Total number of TSCs (#TSCs)</td>
<td>11,842,166</td>
</tr>
<tr>
<td>Ave. TSCs per movie</td>
<td>1,652.5</td>
</tr>
<tr>
<td>Ave. users per movie</td>
<td>465.9</td>
</tr>
<tr>
<td>Ave. TSCs per user</td>
<td>10.4</td>
</tr>
<tr>
<td>Max/Min number of TSCs for a movie</td>
<td>8,028/101</td>
</tr>
<tr>
<td>Max/Min number of users for a movie</td>
<td>3,370/1</td>
</tr>
<tr>
<td>Max/Min number of TSCs for a user</td>
<td>68,236/1</td>
</tr>
</tbody>
</table>
Methodology

- Use neural network based collaborative filtering for modelling user and key frame embeddings.
- Use long-short term memory network for modelling the user comments.
- Integrate the two methods to predict the next key frame.
Matrix factorization using neural network

- Find user and item embeddings using neural network.
Image based model

- Preprocessed image features are merged with the frame latent factors to derive new embedding.
- Which is used to generate prediction.
Image based model

\[ q_k^* = \text{MERGE}(q_k, W^{image} \cdot vsl_k) \]

\[ \text{MERGE}((a_1, a_2, ..., a_K), (b_1, b_2, ..., b_K)) = (a_1 b_1, a_2 b_2, ..., a_K b_K) \]

\[ \hat{y}_{uk}^{image} = \text{LOGISTIC}(p_u \cdot q_k^*). \]
Loss function for image based features

\[ L_1 = \log \prod_{(u,k)} \left( \hat{y}_{uk}^{image} \right)^{y_{uk}} \left( 1 - \hat{y}_{uk}^{image} \right)^{1-y_{uk}} \]

\[ = \log \prod_{(u,k) \in O^+} \hat{y}_{uk}^{image} \prod_{(u,k) \in O^-} (1 - \hat{y}_{uk}^{image}) \]

\[ = \sum_{(u,k) \in O^+} \log \hat{y}_{uk}^{image} + \sum_{(u,k) \in O^-} \log (1 - \hat{y}_{uk}^{image}) \]
Text based model

- Preference embedding is added to every time step in LSTM.

- LSTM captures the words in the comments.
Text based model

\[ h_1 = LSTM(w_{tscuk}^0, e_{uk}^{pre_0}) \]

\[ h_t = LSTM(h_{t-1}, w_{tscuk}^{t-1}, e_{uk}^{pre_0}) \quad t \in \{2, 3, \ldots s_{uk}\} \]

\[ p(w_{tscuk}^t | e_{uk}^{pre_0}, w_{tscuk}^{0:t-1}) = \text{SOFTMAX}(h_t) \]
Loss for Text based model

\[
L_2 = \sum_{(u,k) \in O^+ \cup O^-} \sum_{t=1}^{s_{uk}-1} \log p(w_{tsc_{uk}}^t | e_{uk}^{pre0}, w_{tsc_{uk}}^{0:t-1}) \\
+ \sum_{(u,k) \in O^+} \log \hat{y}_{uk}^{TSC} + \sum_{(u,k) \in O^-} \log (1 - \hat{y}_{uk}^{TSC})
\]
Integrated model: Baseline
Integrated model
Integrated Model

\[ e^{pre0}_{uk} = p_u \odot q_k^* \]

\[ e^{prei}_{uk} = g^{nl}(W^i \cdot e^{prei-1}_{uk}) \quad i \in \{1, 2, \ldots, D\} \]

\[ \hat{y}^{integrated}_{uk} = \text{LOGISTIC}(w^{output} \cdot e^{pred}_{uk}) \]

\[ L_3 = \alpha \sum_{(u,k) \in O^+ \cup O^-} \sum_{t=1}^{s_{uk}-1} \log p \left( w^t_{ts\cdot u_k} \mid e^{preinit}_{uk}, w_{0:t-1} \right) + \]

\[ (1 - \alpha) \left( \sum_{(u,k) \in O^+} \log \hat{y}^{integrated}_{uk} + \sum_{(u,k) \in O^-} \log \left(1 - \hat{y}^{integrated}_{uk}\right) \right) \]
Evaluation method

- If a user comments on a keyframe with positive sentiment, then this frame would be the one that attracts her, so the empirical experiments are conducted by comparing the predicted keyframes with the true positive ones.

- 30% of each user’s positive keyframes are selected as the test dataset, while the others are used for training.
Results
Key takeaways

- Combines model-based collaborative filtering and long-short term memory network together.
- To model user commented keyframes and time synchronized comments simultaneously.
Questions?
The Role of Social Networks in Information Diffusion

(Bakshy et al, 2012)

- Shreyas Udupa Balekudru
Motivation

- Social networks enable individuals to share information simultaneously with a number of peers.
- How to quantify the causal effect of these mediums on the dissemination of information?
- How to identify who influences whom?
- Will individuals propagate information in the absence of social signals about that information?
- Conducting a large-scale field experiment
- What is the relative role of strong and weak ties in information propagation?
Challenges

- Despite availability of social network data, identifying influence is a challenge. Correlation between two individuals’ behaviour may be because they are similar or because one has influenced the other.
- Homophily (tendency of individuals with similar characteristics to associate with one another) makes it difficult to estimate the relative role of strong and weak ties in information propagation. Collecting data on tie strength is prone to biases.
- Most field experiments are confined to studying spread of highly specific information within limited populations.
Facebook is ideal for such a study

- Most widely used social networking service in the world.
- As of 2012:
  - More than 800 million users each month
  - In the US, 54% of adult internet users were on Facebook
  - On average, American users maintained 48% of their real world contacts on Facebook.
- Individuals regularly exchange news items with contacts.
- Interaction among users is well correlated with self-reported intimacy.
Issues to consider

- Diffusion behaviour generally exhibits correlation between number of friends exhibiting the behaviour and probability of adopting the behaviour. It is necessary to differentiate between the effect of homophily versus the effect of peer influence.
- Information does not spread only through social media. To understand importance of social networks in information diffusion, the key is to identify sources of interpersonal contagion and understand what would happen if certain interactions did not take place.
Experimental Design

- Evaluate how much exposure to a URL on the feed increases an individual’s propensity to share that URL.
- All unobservable correlations can be identified by blocking the causal relationship between the Facebook feed and sharing.
Assignment Procedure

- Subject-URL pairs are randomly assigned to ‘feed’ or ‘no feed’ condition.
- Pairs are deterministically assigned.
- To improve statistical power, twice as many pairs were assigned to ‘no feed’ condition.
- A shared URL is on average delivered to over 99% of its potential targets.
- All activity related to these pairs is logged. A pair is removed if there is interaction observed outside the feed.
- 253,238,367 subjects, 75,888,466 URLs and 1,168,633,941 unique subject-URL pairs, seven weeks.
Ensuring Data Quality

- Only content that was shared by the subjects’ friends after the start of the experiment is considered.
- For subject-URL pairs assigned to the ‘no feed’ condition, pairs where interaction has been observed via any interface on the website up to two months prior to exposure are excluded.
- Facebook’s site integrity system is used to classify and remove URLs that may not reflect ordinary users’ purposeful intentions of distributing content to their friends.
Population

- Random sample of users between Aug 14\textsuperscript{th} and Oct 4\textsuperscript{th} 2010, with at least one friend sharing a link.
- Approximately 253 million users.
- All users report gender and age. Country of residence is determined by using the IP address used to access the site.
- Median age – 26 years
- Average age – 29.3 years
- 236 countries / territories (44 with 1 million or more subjects)
<table>
<thead>
<tr>
<th>Demographic Feature (% of subjects)</th>
<th>feed</th>
<th>no feed</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Gender</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FEMALE</td>
<td>51.6%</td>
<td>51.4%</td>
</tr>
<tr>
<td>MALE</td>
<td>46.7%</td>
<td>47.0%</td>
</tr>
<tr>
<td>UNSPECIFIED</td>
<td>1.5%</td>
<td>1.5%</td>
</tr>
<tr>
<td><strong>Age</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17 OR YOUNGER</td>
<td>12.8%</td>
<td>13.1%</td>
</tr>
<tr>
<td>18-25</td>
<td>36.4%</td>
<td>36.1%</td>
</tr>
<tr>
<td>26-35</td>
<td>27.2%</td>
<td>26.9%</td>
</tr>
<tr>
<td>36-45</td>
<td>13.0%</td>
<td>12.9%</td>
</tr>
<tr>
<td>46 OR OLDER</td>
<td>10.6%</td>
<td>10.9%</td>
</tr>
<tr>
<td><strong>Country (top 10 &amp; other)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNITED STATES</td>
<td>28.9%</td>
<td>29.1%</td>
</tr>
<tr>
<td>TURKEY</td>
<td>6.1%</td>
<td>5.8%</td>
</tr>
<tr>
<td>GREAT BRITAIN</td>
<td>5.1%</td>
<td>5.2%</td>
</tr>
<tr>
<td>ITALY</td>
<td>4.2%</td>
<td>4.1%</td>
</tr>
<tr>
<td>FRANCE</td>
<td>3.8%</td>
<td>3.9%</td>
</tr>
<tr>
<td>CANADA</td>
<td>3.8%</td>
<td>3.8%</td>
</tr>
<tr>
<td>INDONESIA</td>
<td>3.7%</td>
<td>3.5%</td>
</tr>
<tr>
<td>PHILIPPINES</td>
<td>2.1%</td>
<td>2.3%</td>
</tr>
<tr>
<td>GERMANY</td>
<td>2.3%</td>
<td>2.3%</td>
</tr>
<tr>
<td>MEXICO</td>
<td>2.0%</td>
<td>2.1%</td>
</tr>
<tr>
<td>226 OTHERS</td>
<td>37.5%</td>
<td>37.7%</td>
</tr>
</tbody>
</table>
Evaluating Outcomes

- Causal effect of exposure via feed on sharing = expected probability of sharing in ‘feed’ condition – expected probability of sharing in ‘no feed’ condition
- This is known as ‘Average Treatment Effect on the Treated’ or as ‘Absolute Risk Increase’.
- It can also be viewed as a ratio instead of as a difference.
- Active users and popular URLs may introduce bias. Bootstrapped averages clustered by the URL are used to provide control.
Exposure to Social Signals affects Diffusion

- Likelihood of sharing in ‘feed’ condition = 0.191%
- Likelihood of sharing in ‘no feed’ condition = 0.025%
- Upon exposure, an individual is 7.37 times more likely to share.
- Above numbers are significant because hundreds may see the link if shared and 1 in 12.5 links that are clicked on are re-shared.
Temporal Clustering

- Median sharing latency after a friend has already shared the content is 6 hours in the ‘feed’ condition, compared to 20 hours when assigned to the ‘no feed’ condition.
- Presence of strong temporal clustering in both experimental conditions.
- Regardless of access to social signals within a particular online medium, individuals can still acquire and share the same information as their friends, albeit at a slightly later point in time.
Effects of Multiple Sharing Friends

Social information in the feed is most likely to influence a user to share a link that many of her friends have shared, but the relative impact of that influence is highest for content that few friends are sharing.
Tie Strength and Influence

Tie strength measured in terms of interactions:
- Frequency of private online communication (Facebook messages)
- Frequency of public online communication (comments on another’s posts)
- Number of real-world coincidences (appearing in the same photograph)
- Number of online coincidences (both commenting on same post)
Tie Strength and Influence
Tie Strength and Influence

Strong ties are individually more influential, but the effect of strong ties is not large enough to match the sheer abundance of weak ties.
Questions?
Online Popularity and Topical Interests through the Lens of Instagram (Ferrara et al., 2014)

Presenter: Sejal Shah
Background

- Online socio-technical systems are a proxy to real world behavior
- Instagram (had 100M+ users in 2014) represents a mixture of features of various social media and online social networks
  - Creating user generated content in form of visual media
  - Social tagging, establishing follower-followee relationships
  - Social interactions like commenting and liking
- The paper addresses research questions spanning different areas of network, semantic, and topical based data analysis using signals from user activities and interactions
Research Questions

● **Network and community structure:**
  ○ What are the salient structural features in the network built on the users' interactions?

● **Content production and consumption:**
  ○ How do users get engaged on the platform and interact with content by others?

● **Social tagging:**
  ○ How diverse is the set of tags exploited by each user?

● **Topical clusters of interest:**
  ○ How can users be grouped based on the tags they use to annotate media?

● **Popularity and topicality:**
  ○ How does the topical interests of users affect their popularity?
Data Preparation

- Instagram sample by querying public IG API such as users, relationships, media, comments and likes, and tags APIs
- Crawling based on retrieving users that belong to a relatively large “community” in Instagram
  - Instagram does not offer an explicit group/community feature
  - Exploited a public initiative by IG called Weekend Hashtag Project (#whp)
  - Selected 72 popular contests and randomly picked 2,100 users (all their media, likes, comments, tags)
- Collected over one month, 1.7M media with 9M tags, 1.2B likes and 41M comments
- Limitation:
  - Competition driven dataset collection might not reflect actual behaviour
  - Latent interactions (profile browsing) are not captured
Relational Instagram Network

- Directed weighted graph
  - Edges are modeled to reflect asymmetric relationships of follower-followee
  - Edge weights are proportional to likes and comments
  - Seed nodes are obtained from the users selected to build the media dataset

<table>
<thead>
<tr>
<th>Relational Instagram network statistics.</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. Nodes</td>
</tr>
<tr>
<td>No. Links</td>
</tr>
<tr>
<td>Avg. In-degree</td>
</tr>
<tr>
<td>Avg. Path length</td>
</tr>
<tr>
<td>Clustering coefficient</td>
</tr>
<tr>
<td>Diameter</td>
</tr>
<tr>
<td>Assortativity index</td>
</tr>
<tr>
<td>No. Communities</td>
</tr>
<tr>
<td>Network modularity</td>
</tr>
</tbody>
</table>
1. **Structural features of the IG network**

What are the salient structural features in the network built on the users' interactions?

- The aim is to study the topological features and determine if a social process exists
  - Whether the topical interests of the users have any effect on the network structure
- Community detection carried out using *Louvain* method and OLSOM (consistent results)
- Broad distributions for node degree and community size suggest RIN growth may follow a preferential-attachment mechanism
- Based on the formation of communities of heterogeneous size, emergence of self organization is suggested
- Further, Q2 and Q3 analyse whether these communities are topically-induced structures
I. Community structure and II. Distribution of node degree and community size of the Relational Instagram Network
2. Content production

How do users get engaged on the platform and interact with content by others?

- PDF showed peculiar content production dynamics on IG:
  - Users exhibit higher tendency to post new content if they already did in the past
  - Clear separation between active and inactive users
  - Strategy could be used in designing and implementing new features to engage inactive users to lower imbalance in users involved in content production
I. User content production distribution

II. Social interactions distribution
2. Content consumption

- Two consumption dynamics, namely “like” and comment, have been analyzed.
- PDF suggests that the consumption on Instagram might follow two different dynamics:
  - Popularity of media measured by number of likes grows by preferential attachment (media with large number of likes are more likely to acquire even more)
  - Ecosystem is less prone to trigger large conversations (based on comments)
  - Associated with different costs (e.g., in terms of time required to perform the action) between “liking” some content and writing a comment affect the nature of interactions among individuals.
3. Social tagging dynamics

*How diverse is the set of tags exploited by each user?*

Investigated three aspects:

- Tag popularity at the global level and the distribution of tags per media
  - Majority of media are labeled with just a few tags
  - Tag popularity follows a power law behaviour
- Distribution of total tags used by the users and their vocabulary size
  - Actual user vocabulary size is limited, a large majority of users adopt only few tags
- Diversity in tag usage (continued on next slide)
I. Tag adoption and global popularity

II. Tag usage and tagset size dist.

\[ \gamma = 1.865 \]
3. Diversity in tag usage

- Tagging entropy is a proxy to measure how spread or focused users' attention is towards few or several contexts.
- The lower the entropy, the more focused a user’s tagging pattern is, the more diverse is her/his tagging behavior.
- ~50% users exhibit an average tagging variety.
4. Topical clusters of interest

How can users be grouped based on the tags they use to annotate media?

- Users represented as tf vectors in the space of media tags
- Hard clustered using bisecting k-means; best solution at k = 5
- Only the most descriptive and discriminating features are included in the plot
5. Topicality

- First performed topic modeling using Latent Semantic Indexing, number of topics = 10
- Applied topical entropy similar to tagging entropy seen earlier
  - Topical entropy is concentrated between 2.5 to 3.5 as opposed to 0 to 9 for tags
  - Although users are equally likely to adopt either a narrow or broad vocabulary of tags, their topical interests tend to be in general more concentrated
5. User popularity

- Popularity is total number of likes and comments received by a user’s media.
- Social actions are total likes and comments by the user on others’ media.
- Steeper slope of social actions suggests relatively less users who produce many likes and comments.
5. User popularity and topicality

How does the topical interests of users affect their popularity?

- As popularity grows, the topical entropy increases accordingly
- Presence of popular users with topical entropy much lower or much higher than average
Conclusion

- **Network and community structure**: Topical interests exhibited by users might affect their inter-connectivity and interactions.
- **Content production & consumption**: Strong heterogeneity in the mechanism of production of new information; emergence of information economy principle in content consumption.
- **Social tagging behavior**: Users exhibit vocabularies of limited size, nonetheless, popular trends emerge.
- **Topicality**: Clusters of users can be found around tags.
- **Topics and popularity**: Users with narrow interests tend to be less popular, whereas broader interests yield higher popularity.
- Popular users are special with more extreme behavior: they produce either very topically specific content, or media of very broad interest.
Conclusion

- Broad analysis of Instagram ecosystem, exploiting its heterogeneous structure, part social network, part tagging environment, and part media sharing platform
- Used user relationships and interactions to perform network-, semantic- and topical- based analysis on users, media and how these two dimensions are interrelated
- Further work could assess role of network structure in popularity of content and users
- Unavailability of clickstream data through Instagram API creates limitations on analysing user latent interactions
Thank you!
Exploiting Socio-Economic Models for Lodging Recommendation in the Sharing Economy
(Vazquez et al. 2017)

Presented by - Prem Nagarajan
Introduction

- The motivation behind this paper is to recommend lodging to users.
- The aim is to consider the taste and preferences of the user as well as other socio-economic factors while making recommendations.
- The number of users who have made more than 5 reservations in Airbnb is really low.
- The Airbnb user profiles are severely sparse.
- Limits the applicability of traditional collaborative or content based approaches.
Profile size distribution of Airbnb users
Proposed model

- Context aware learning-to-rank approach for lodging recommendation (CLLR)
- Five broad aspects to consider
  1. **Perceived Value (PV)**: the trade-off between the benefits versus the cost of each available lodging
  2. **Perceived Risk (PR)**: the assessment of all possible negative outcomes derived from booking the lodging
  3. **Price Sensitivity (PS)**: the extent to which the price of a lodging affects a guest’s booking behavior
  4. **Perceived Authenticity (PA)**: the extent to which a guest feels like natively living at the lodging place
  5. **Electronic-Word-of-Mouth (EWoM)**: informal opinions that frame the judgment of other users towards the lodging
<table>
<thead>
<tr>
<th>feature class</th>
<th>input qty</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Perceived Value (PV)</strong></td>
<td></td>
</tr>
<tr>
<td>Pricing</td>
<td>i 6</td>
</tr>
<tr>
<td>Property type</td>
<td>i 21</td>
</tr>
<tr>
<td>Room type</td>
<td>i 3</td>
</tr>
<tr>
<td>Bed type</td>
<td>i 5</td>
</tr>
<tr>
<td>Equipments</td>
<td>i 4</td>
</tr>
<tr>
<td>Property capacity</td>
<td>i 1</td>
</tr>
<tr>
<td>Guests allowed</td>
<td>i 1</td>
</tr>
<tr>
<td>Amenities</td>
<td>i 40</td>
</tr>
<tr>
<td>Nearby venues</td>
<td>c 3</td>
</tr>
<tr>
<td>Nearby venues check-ins (min, max, avg, med)</td>
<td>c 12</td>
</tr>
<tr>
<td>Nearby venues distance (min, max, avg, med, norm.)</td>
<td>i, c 24</td>
</tr>
<tr>
<td><strong>Perceived Risk (PR)</strong></td>
<td></td>
</tr>
<tr>
<td>Cancellation policy</td>
<td>i 5</td>
</tr>
<tr>
<td>Ratings</td>
<td>i 7</td>
</tr>
<tr>
<td>Reviews (std, norm.)</td>
<td>i 2</td>
</tr>
<tr>
<td>Nearby lodgings</td>
<td>c 1</td>
</tr>
<tr>
<td>Nearby lodgings reviews (avg, std)</td>
<td>c 2</td>
</tr>
<tr>
<td><strong>Price Sensitivity (PS)</strong></td>
<td></td>
</tr>
<tr>
<td>Histogram lodgings prices (avg, skw, kur)</td>
<td>c 3</td>
</tr>
<tr>
<td>Sampled lodgings prices (avg, skw, kur)</td>
<td>c 3</td>
</tr>
<tr>
<td>Price (normalized)</td>
<td>i, c 3</td>
</tr>
<tr>
<td><strong>Perceived Authenticity (PA)</strong></td>
<td></td>
</tr>
<tr>
<td>Authenticity score (avg, med, min, max, skw, kur)</td>
<td>i 6</td>
</tr>
<tr>
<td><strong>Electronic Word of Mouth (EWoM)</strong></td>
<td></td>
</tr>
<tr>
<td>Sentiment score (avg, med, min, max, skw, kur)</td>
<td>i 24</td>
</tr>
<tr>
<td><strong>Grand total</strong></td>
<td>176</td>
</tr>
</tbody>
</table>
Datasets

- Airbnb data for New York City and London
- Between March and September 2016
- Lodgings spread over the city

<table>
<thead>
<tr>
<th></th>
<th>NYC</th>
<th>LON</th>
<th>World</th>
<th>Airbnb</th>
</tr>
</thead>
<tbody>
<tr>
<td># Guests</td>
<td>219.9 k</td>
<td>223.1 k</td>
<td>9.25 M (31.9%)</td>
<td>60.0 M</td>
</tr>
<tr>
<td># Lodgings</td>
<td>17.3 k</td>
<td>22.1 k</td>
<td>0.48 M (26.3%)</td>
<td>2.0 M</td>
</tr>
<tr>
<td># Transactions</td>
<td>250.5 k</td>
<td>266.7 k</td>
<td>15.70 M (15.4%)</td>
<td>49.2 M</td>
</tr>
</tbody>
</table>
Questions being addressed

- How effective is our lodging recommendation approach?
- How do different features contribute to our approach?
- How do our results relate to lodging consumption theories?
Baselines models

- **Popularity** is a bias-centric model based on the lodging’s booking count. Ties are solved by randomly sorting conflicting items.
- **Airbnb** consists of the non-personalized ranking produced by Airbnb’s own recommender given the same input as all the other recommenders in our evaluation.
- **Bayesian Personalized Ranking Matrix Factorization (BPRMF)** is a state-of-the-art matrix factorization approach for top-n recommendation.
Contextual Learning-to-rank approach for Lodging Recommendation (CLLR)

- Considers a total of 176 features organized into five broad preference dimensions.
- Goal is to learn a ranking model \( f : X \rightarrow Y \) mapping the input space \( X \) into the output space \( Y \).
- Input space includes \( n \) learning instances \( \{X_j\}_{j=1}^n \), where \( X_j = \Phi(u_j, l_j, I_j) \) is a feature matrix representation of a sample of lodgings retrieved for user \( u \) near target location \( l \).
- Output space \( Y \) comprises \( n \) label vectors \( \{Y_j\}_{j=1}^n \), where \( Y \) provides relevance labels for each lodging.
- Uses LambdaMART, a gradient boosted regression tree learner, which represents the current state-of-the-art in learning to rank.
Mean Reciprocal Rank

<table>
<thead>
<tr>
<th></th>
<th>BPRMF</th>
<th>Popularity</th>
<th>Airbnb</th>
<th>CLLR</th>
</tr>
</thead>
<tbody>
<tr>
<td>MRR</td>
<td>0.0215</td>
<td>0.0231</td>
<td>0.0328</td>
<td>0.0400</td>
</tr>
<tr>
<td>CI (95%)</td>
<td>0.0014</td>
<td>0.0015</td>
<td>0.0018</td>
<td>0.0021</td>
</tr>
</tbody>
</table>
MRR over a time window
Effect of individual dimensions on MRR

Table 5: MRR after removing individual dimensions.

<table>
<thead>
<tr>
<th></th>
<th>PV</th>
<th>PR</th>
<th>PS</th>
<th>PA</th>
<th>EWoM</th>
</tr>
</thead>
<tbody>
<tr>
<td>MRR</td>
<td>0.0383</td>
<td>0.0418</td>
<td>0.0412</td>
<td>0.0397</td>
<td>0.0400</td>
</tr>
<tr>
<td>CI (95%)</td>
<td>0.0020</td>
<td>0.0022</td>
<td>0.0021</td>
<td>0.0020</td>
<td>0.0020</td>
</tr>
<tr>
<td>Feature</td>
<td>(EFE/LSI)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Perceived Value (PV)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dist. Mean (Travel)</td>
<td>7.1°/0.001</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dist. Max (Food)</td>
<td>5.7°/0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dist. Mean (Food)</td>
<td>4.3°/0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security Dep.</td>
<td>3.9°/0.003</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Check-ins Mean (Arts)</td>
<td>3.5°/0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Check-ins Min. (Arts)</td>
<td>2.9°/0.004</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Check-ins Med. (Travel)</td>
<td>2.8°/0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Check-ins Min. (Food)</td>
<td>2.7°/0.017</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bedrooms</td>
<td>0.6°/0.002</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pets Allowed</td>
<td>-2.0°/0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hair Dryer</td>
<td>-2.1°/0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dist. Norm Max (Food)</td>
<td>-2.2°/0.017</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Airbed (Bed)</td>
<td>-2.2°/0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pool</td>
<td>-2.9°/0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Price</td>
<td>-3.1°/0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dist. Med. (Arts)</td>
<td>-3.3°/0.003</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dist. Min (Food)</td>
<td>-4.5°/0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| Perceived Risk (PR)  | 8.1°/0.018 |
| Rev. Cnt. Norm. (\(\bar{r}\)) | 5.4°/0.001 |
| Context Rev. Cnt. Std. (\(\sigma_r\)) | 4.0°/0.006 |
| Star Rating           | 3.0°/0.001 |
| Location (Rating)     | -9.8°/0.01   |

<table>
<thead>
<tr>
<th>Perceived Sensitivity (PS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Price Kurtosis</td>
</tr>
<tr>
<td>Airbnb Skewness</td>
</tr>
<tr>
<td>Price Norm. Airbnb ((p_{\text{AR}}))</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Electronic Word of Mouth (EWoM)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comp. Skewness</td>
</tr>
<tr>
<td>Pos. Skewness</td>
</tr>
<tr>
<td>Neg. Med.</td>
</tr>
<tr>
<td>Neg. Mean</td>
</tr>
<tr>
<td>Neu. Med.</td>
</tr>
<tr>
<td>Comp. Med.</td>
</tr>
<tr>
<td>Comp. Max</td>
</tr>
<tr>
<td>Pos. Min</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Perceived Authenticity (PA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auth Min.</td>
</tr>
<tr>
<td>Auth Kurtosis</td>
</tr>
</tbody>
</table>
Structural Equation Models (SEM) for Repurchase Intention
Conclusion

- CLLR performs better than state of the art CF techniques
- Customers’ **perceived risk** is one of the most important factors that impacts consumption behavior, which also affects customers’ **perceived value** of a lodging
- Future work involves investigating alternative sources of contextual information in order to improve the power of the features
Thank you