# Group Recommender Systems

## Rank Aggregation and Balancing Techniques

Linas Baltrunas, Tadas Makcinskas,
Auste Piliponyte, **Francesco Ricci**

Free University of Bozen-Bolzano

Italy

fricci@unibz.it

# Content

- Group recommendations
- Rank aggregation – optimal aggregation
- Rank aggregation for group recommendation
- Dimensions considered in the study
  - Group size
  - Inter group similarity
  - Rank aggregation methods
- Sequential Group Recommendations
- Balancing
- User study

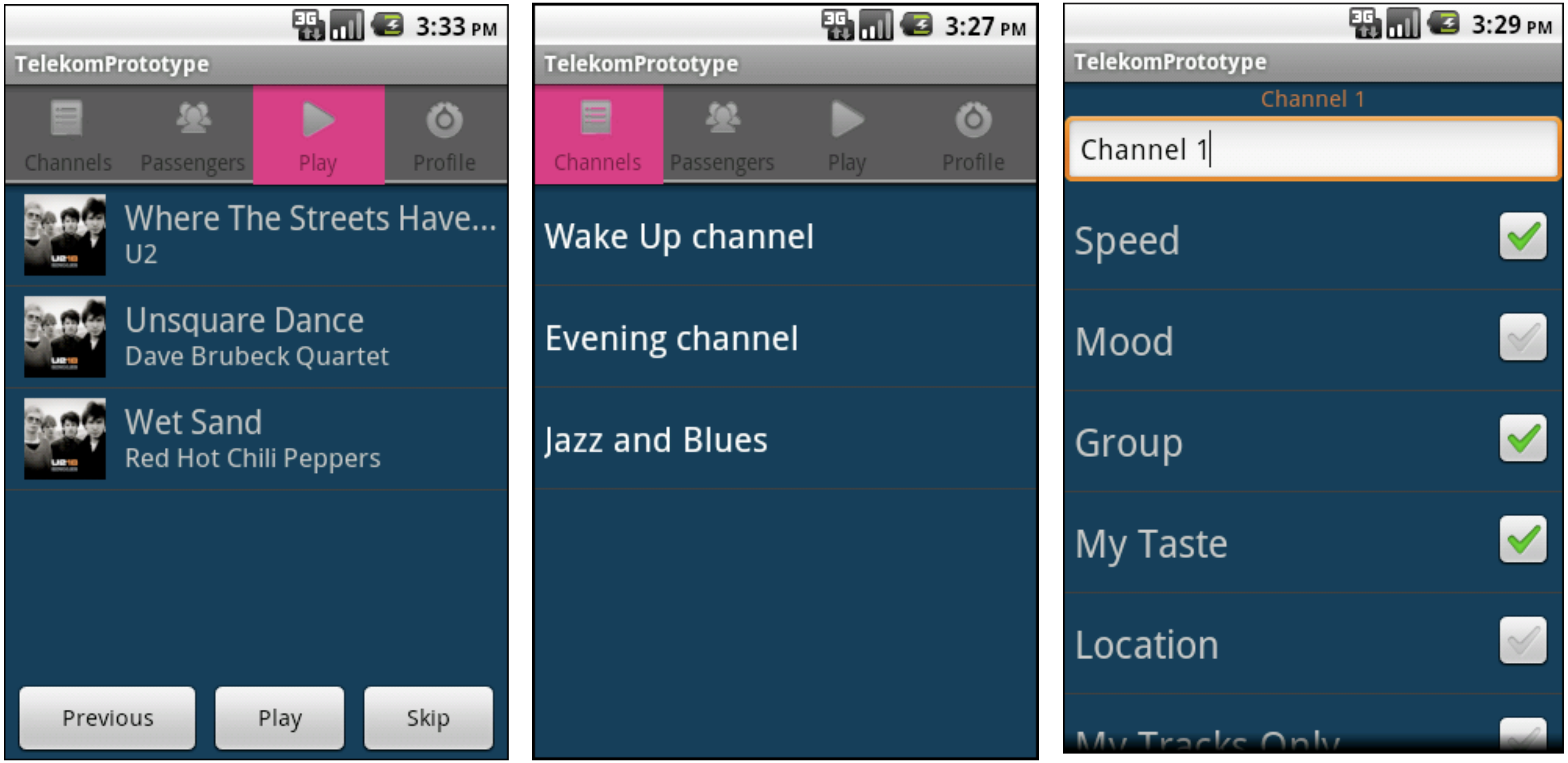
# Group Recommendations

- Recommenders are usually designed to provide recommendations **adapted** to the **preferences** of a **single user**
- In many situations the recommended items are consumed by a **group of users**
  - A travel with friends
  - A movie to watch with the family during Christmas holidays
  - Music to be played in a car for the passengers

# Mobile Application

Deutsche Telekom Laboratories

- Recommending music compilations in a car scenario




[Baltrunas et al., 2011]

# Group Recommendation Model

- Items will be experienced by individuals **together** with the other group members: the evaluation function **depends** on the **group:**

$$r : U \times I \times \wp(U) \rightarrow E$$

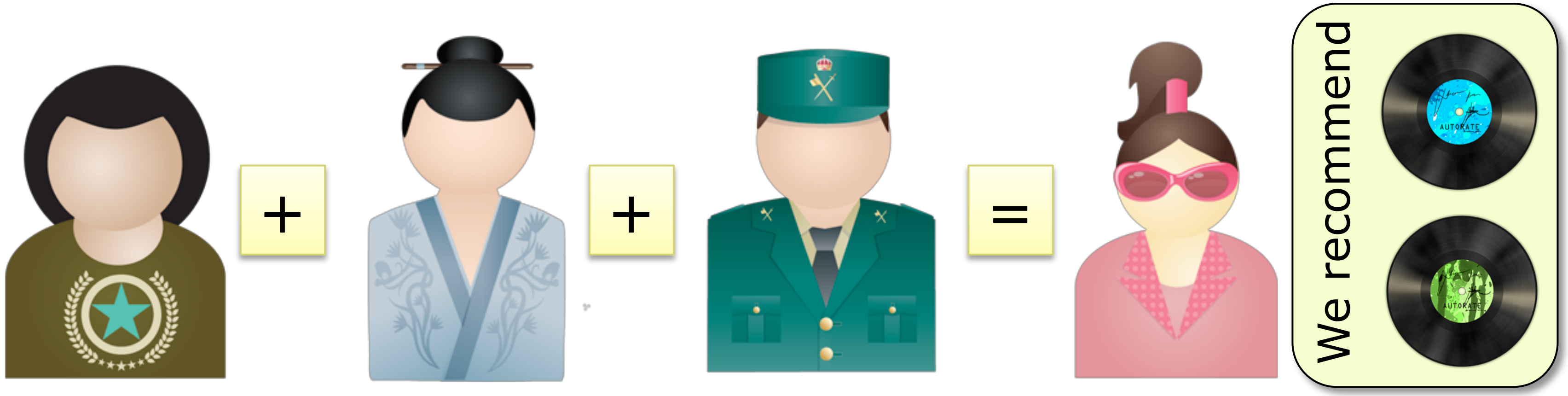
- *U* is the set of users, *I* is the set of Items, *P(U)* is the set of subsets of users (groups), *E* is the evaluation space (e.g. the ratings *{?, 1, 2, 3, 4, 5}*) of the rating function *r*
- Normally researchers assume that *r(u,i)=r(u,i,g)* for all groups $g \ni u$
- But users are influenced in their evaluation by the group composition (e.g., emotional contagion [Masthoff & Gatt, 2006]).

# Recommendation Generation

- Having identified the best items for each group member **how we select the best items for the group?**
- How the concept of "best items" for the group can be defined?
- We could introduce a fictitious user *g* and be able to estimate *r(g,i)*
- But how?
- Two approaches have been considered [Jameson & Smyth, 2007]
  - **Profiles aggregation**
  - **Recommendations aggregation**

# First Mainstream Approach

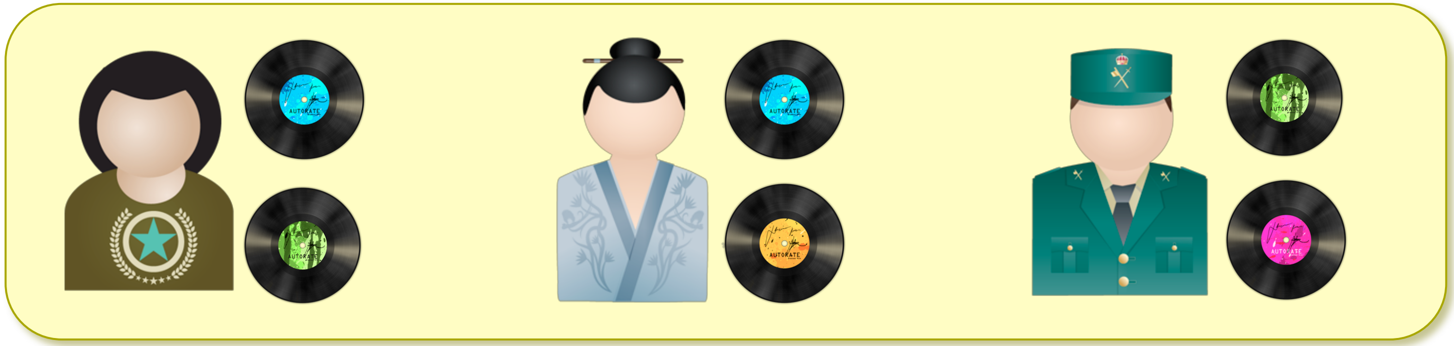
- Creating the **joint profile** of a group of users




- We build a recommendation for this "average" user
- **Issues**
  - The recommendations may be difficult to explain – individual preferences are lost
  - Recommendations are customized for a "user" that is not in the group
  - There is no well founded way to "combine" user profiles – why averaging?

# Second Mainstream Approach

- Producing individual recommendations



- Then “aggregate” the recommendations:

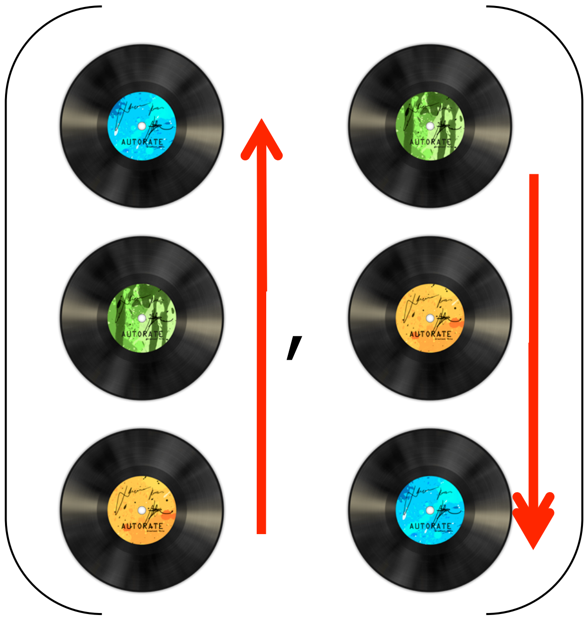
- **Issues**
  - How to optimally aggregate ranked lists of recommendations?
  - Is there any “best method”?

# Optimal Aggregation

- Paradoxically there is not an optimal way to aggregate recommendations lists (Arrows' theorem: *there is no fair voting system*)
- [Dwork et al., 2001] introduced the notion of **Kemeny-Optimal** aggregation:
  - Given a distance function between two ranked lists (Kendall tau distance)
  - Given some input ranked lists to aggregate
  - Compute the ranked list (permutation) that **minimize the average distance** to the input lists.

# Kendall tau Distance

- The number of pairwise disagreements

dist  = 2

↑ One item is preferred to the other

# Kemeny Optimal Aggregation

- Kemeny optimal aggregation is expensive to compute (NP hard – even with 4 input lists)
- There are other methods that have been proved to approximate the Kemeny-optimal solution
  - **Borda count** – no more than 5 times the Kemeny distance [Dwork et al., 2001]
  - **Spearman footrule distance** – no more than 2 times the Kemeny distance [Coppersmith et al., 2006]
    - SFD: the sum over all the elements of the lists of the absolute difference of their rank
  - **Average** – average the predicted ratings and sort
  - **Least misery**- sort by the min of the predicted ratings
  - **Random** – 0 knowledge, only as baseline.

# Average Aggregation

- Let *r*(u,i)* be either the predicted rating of *u* for *i*, or *r(u,i)* if this rating is present in the data set
- Then the score of an item for a group *g* is
  - $r^*(g,i) = AVG_{u \in g} \{r^*(u,i)\}$
- Items are then sorted by decreasing value of their group scores *r*(g, i)*
- ***Issue:*** *the recommended items may be very good for some members and less convenient for others*
- Hence … least misery approach

# Borda Count Aggregation

- Each item in the ranking is assigned a **score** depending on its position in the ranking: the higher the rank, the larger the score is
- The last item $i_n$ in the ranking of user $u$ has $score(u,i_n) = 1$ and the first item has $score(u,i_1) = n$
- **Group score** for an item is calculated by adding up the item scores for each group member:

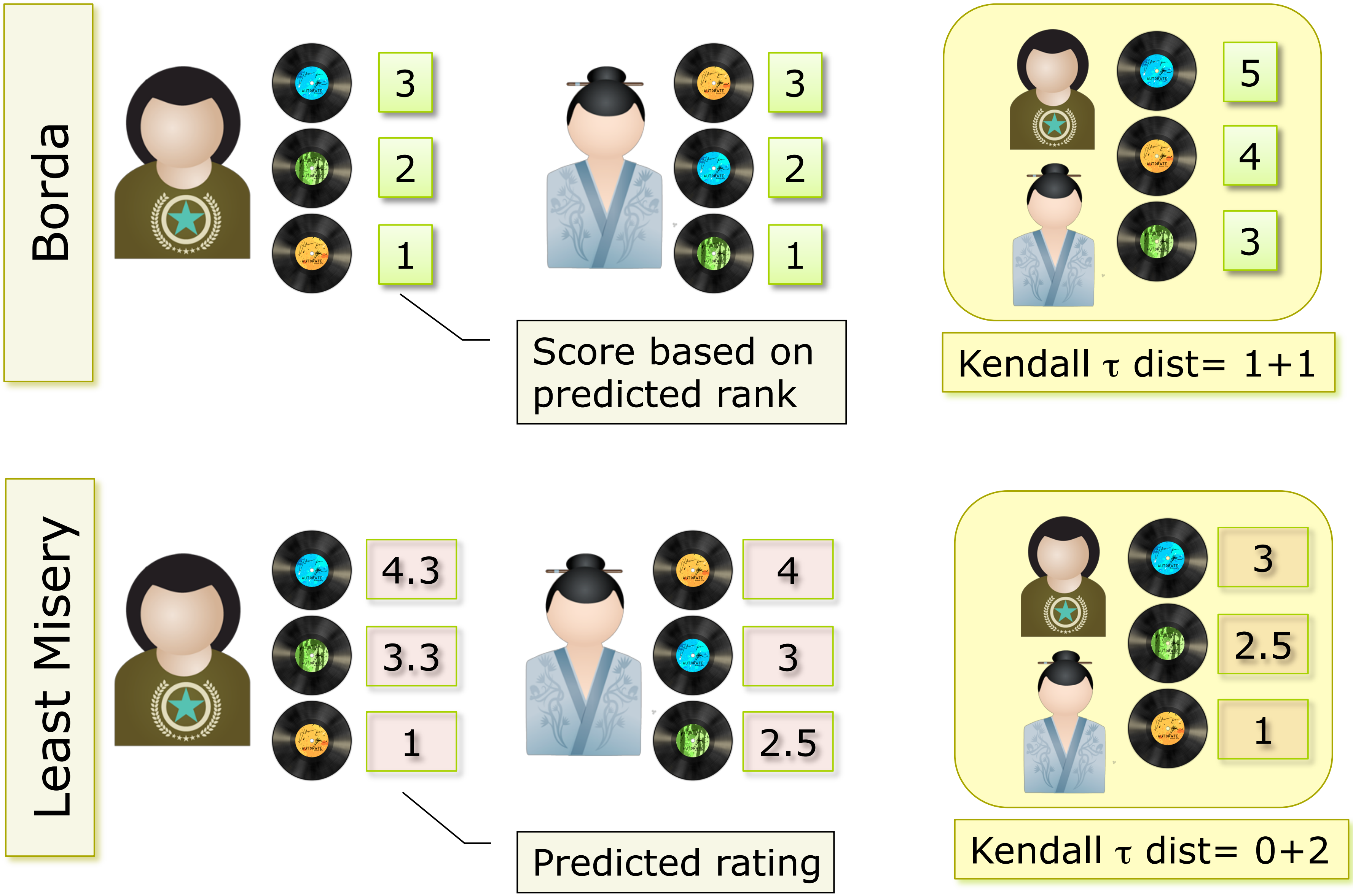
$$score(g,i) = \sum_{u \in g} score(u,i)$$

- Items are then ranked according to their group score.

# Least Misery Aggregation

- Let $r^*(u, i)$ be either the predicted rating of *u* for *i*, or $r(u, i)$ if this rating is present in the data set
- Then the score of an item for a group g is:
  - $r^*(g, i)=MIN_{u \in g} \{r^*(u, i)\}$
- Items are then sorted by decreasing value of their group scores $r^*(g, i)$
- *The recommended items have rather large predicted ratings for **all** the group members*
- *May select items that nobody hates but that nobody really likes (shopping mall case).*

# Borda Count vs. Least Misery



# Evaluating Group Recommendations

- Ask the users to **collectively** evaluate the group recommendations
- Or use a test set for off-line analysis:
  - But how to compare this best "group recommendation" with the **true** "best" item for the group?
  - *What is the ground truth?*
- We need again an aggregation rule that computes the **true** group score for each recommendation
  - $r(g,i) = Agg(r(u_1, i) , \ldots, r(u_{|g|}, i))$
  - $u_i \in g$
- How to define Agg?

# Circular Problem

- If *the* aggregation function used in the evaluation is the same used in the recommendation generation step we have "incredibly" good results

- Example
  - If the items with the largest average of the predicted ratings $AVG_{u \in g}$ {r*(u,i)} are recommended
  - Then these will score better (vs. items selected by a different aggregation rule) if the "true best" recommendations are those with the largest average of their true ratings $AVG_{u \in g}$ {r(u,i)}

# Evaluating Group Recommendations

- Our approach [Baltrunas, Mackcinskas, Ricci, 2010]
- Given a group of users including the active user
- Generate **two ranked lists** of recommendations using a prediction model (matrix factorization) and some training data (ratings):
  a) Either based only on the active user **individual preferences**
  b) Or **aggregating** recommendation lists for the **group of users** (including the active user)
- Compare the recommendation list with the “true” preferences as found in the **test set** of the user
- We have used Movielens data
- Comparison is performed using Normalize Discounted Cumulative Gain.

# Normalised Discounted Cumulative Gain

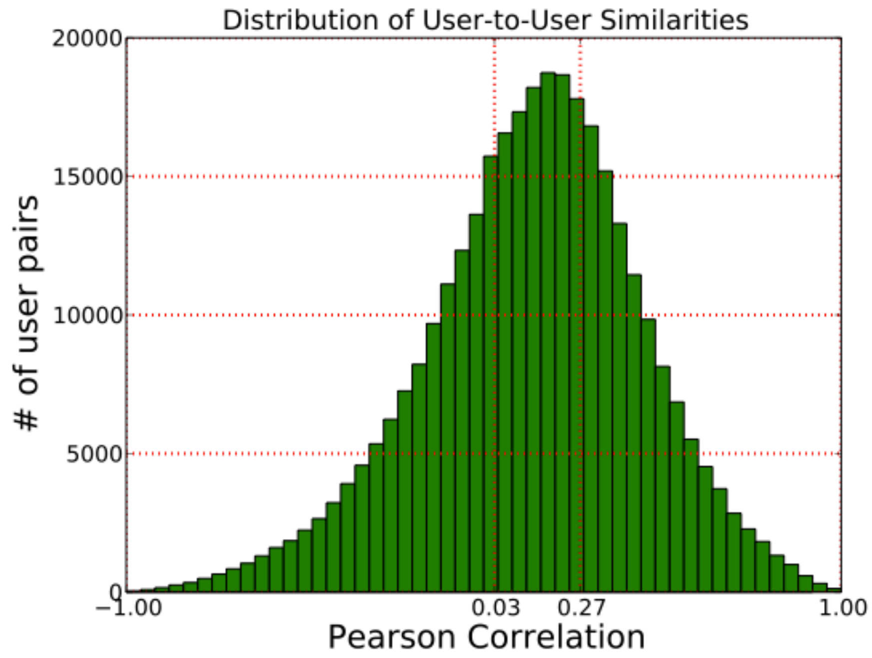
- It is evaluated over the *k* items that are present in the user's test set

$$nDCG_k^u = \frac{1}{Z_{uk}} \sum_{i=1}^{k} \frac{r_{up_i}}{\log_2(i+1)}$$

- $r_{upi}$ is the rating of the item in position i for user u – as it is found in the test set
- $Z_{uk}$ is a normalization factor calculated to make it so that a perfect ranking's NDCG at *k* for user *u* is 1
- It is maximal if the recommendations are ordered in decreasing value of their true ratings.

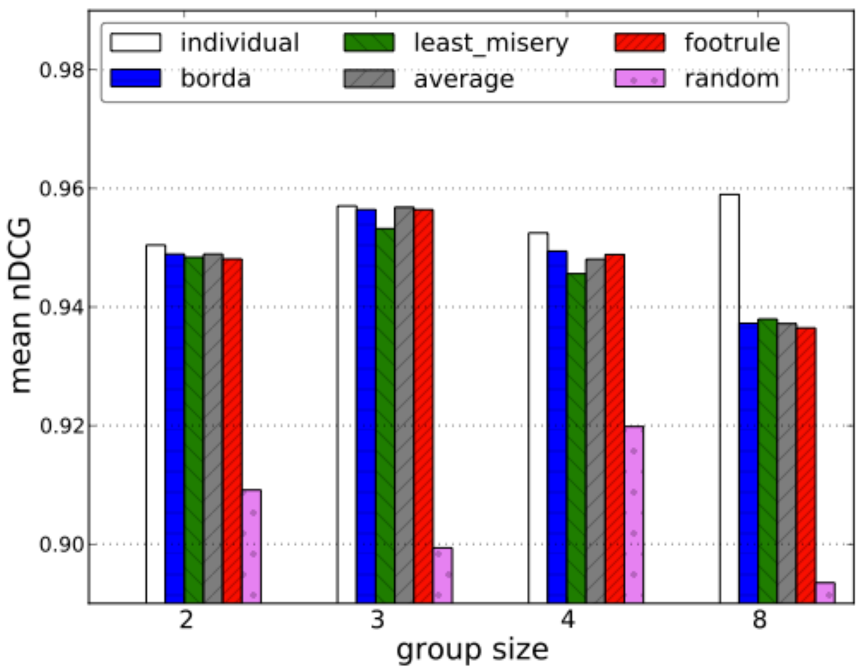
# Building pseudo-random groups

- Groups with **high inner group similarity**
- Each pair of users has Pearson correlation larger than 0.27
- One third of the users' pairs has a similarity larger that 0.27
- We built groups with: 2, 3, 4 and 8 users

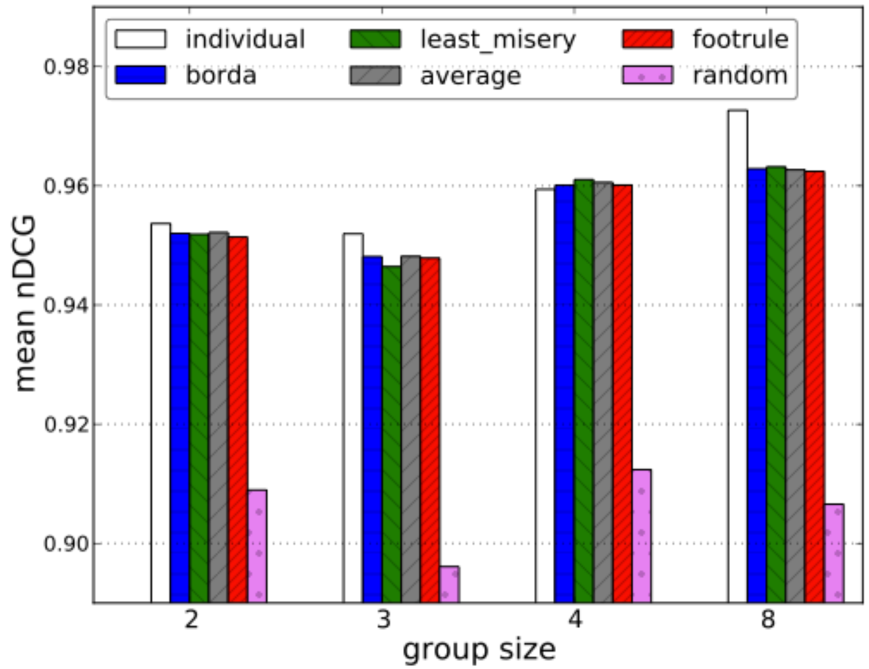



Similarity is computed only if the users have rated at least 5 items in common.

# Random vs Similar Groups




Random Groups




High Inner Group Sim.

- For each experimental condition – a bar shows the average over the users belonging to 1000 groups
- Training set is 60% of the MovieLens data

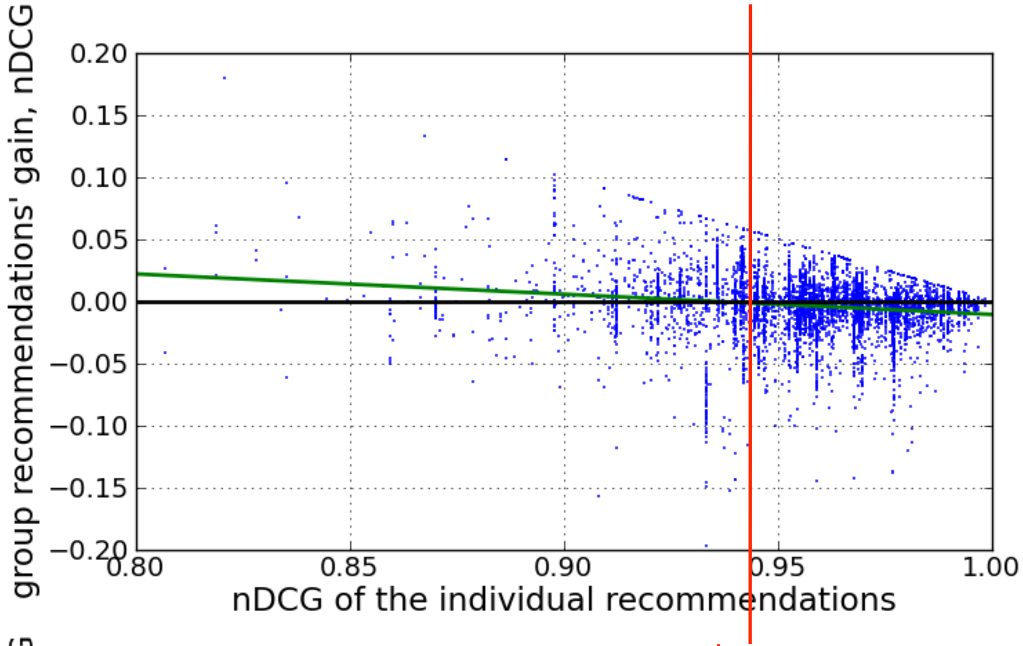
## Group Recommendation Gain

- Is there any **gain** in effectiveness (NDCG) if a recommendations is built for the group the user belongs to?

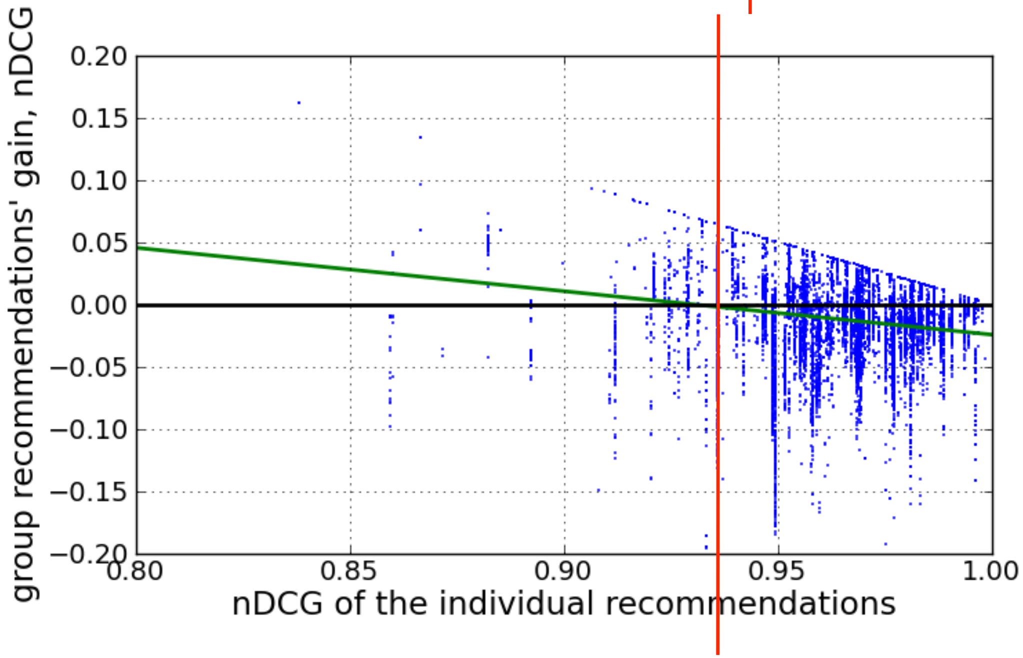
  Gain(u,g) = NDCG(Rec(u,g)) – NDCG(Rec(u))

- When there is a positive gain?
  - Does the quality of the individual recommendations matter?
  - Inner group similarity is important?
- Can a group recommendation be better (positive gain) than an individually tailored one?

# Effectiveness Gain: Individual vs. Group

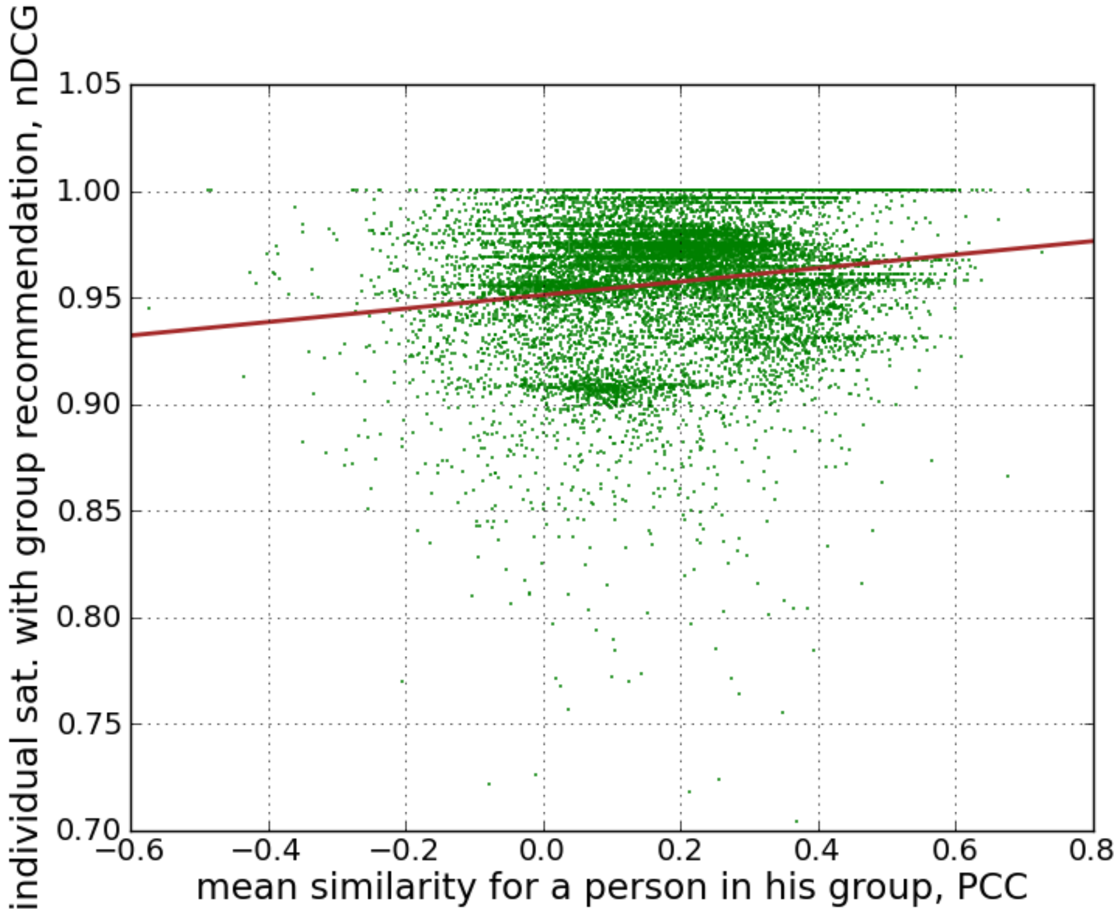



- 3000 groups of 3 users
- High similar users
- Average aggregation




- 3000 groups of 8 users
- High similar users
- Average aggregation

# Effectiveness vs. Inner Group Sim




- Random groups, 4 users
- Average aggregation method

- The larger the inner group similarity is the better the recommendations are – as expected.

## Sequential Recommendations

- How these techniques tackle **sequential** recommendation problems?
- The goal is to compile a sequence of recommendations that receive a large evaluation as a whole
- *Examples:*
  - A sequence of songs
  - A sequence of meals – for the next week
  - A sequence of movies – one for each time a group of friends will meet

# Facets of Sequential Recommendations

- One can re-use the previous techniques and select the top-N recommendations to generate a sequence of length N
- But a sequence of recommendations can be built using other heuristics:
  - *The recommendations should go well together in a given sequence: e.g., uniform mood or genre*
  - *If a user is not totally satisfied with one element of the sequence then he can be made happier with a next element*
  - *User satisfaction for an item is influenced by the previous items (aggregated satisfaction) [Mastoff & Gatt 2006]*
- The recommended sequence must be evaluated as a single recommendation.

# Interface: initial track rating


Thank you for registering.

This web application is providing music track recommendations. Recommendations can be made either individually or for a group of people that would like to listen to music together.

In order to make good recommendations we need you to leave as many track ratings as possible. To begin with we ask you to rate 30 tracks. The tracks can be rated in multiple sessions. You can leave the page and return as many times as you like.

You have rated 6 songs, 24 more left!

Artist: Guns N' Roses

Title: Sweet Child O' Mine

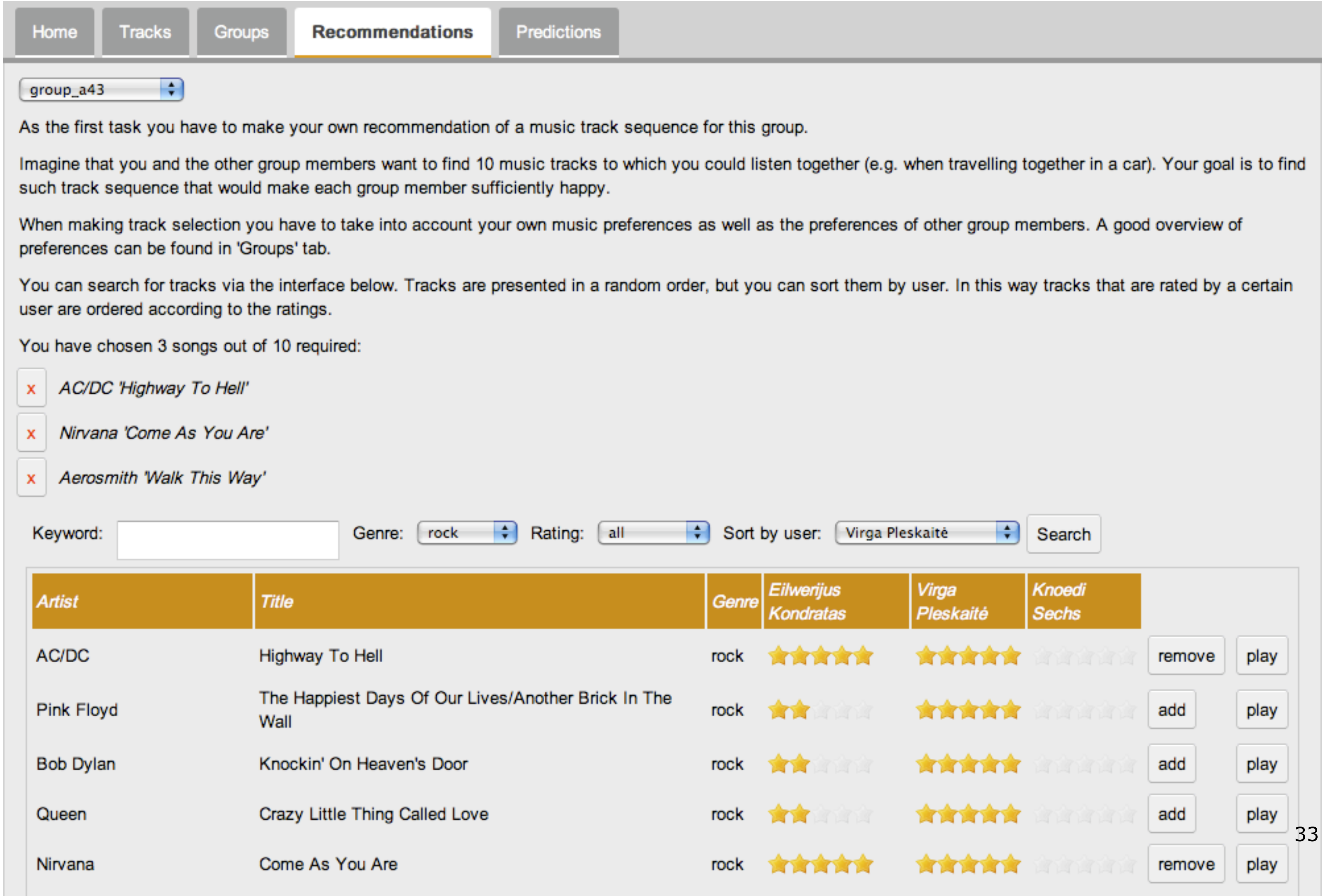
Genre: rock

- * (very bad)
- **
- ***
- ****
- ***** (very good)

Next

[Piliponyte, 2012]

# Interface: recommendation making



Home | Tracks | Groups | **Recommendations** | Predictions

group_a61

This is the last phase of the experiment. We ask you to evaluate two recommendations for music track sequences that were made for your group. One of the recommendations is made by one of your group members and the other by the program. The two recommendations are put in a random order.

When evaluating the recommendations take into consideration the fact that recommendations were made for the whole group and should sufficiently satisfy each group member.

## Recommendation 1:

| Artist | Title | Genre | |
|---|---|---|---|
| Nirvana | Come As You Are | rock | play |
| Green Day | Holiday | pop | play |
| John Mayall & The Bluesbreakers | Kokomo | blues | play |
| Rockmafia | The Big Bang | pop | play |
| Linkin Park | Numb | pop | play |
| Tom Petty | I Won't Back Down | rock | play |
| Jet | Are You Gonna Be My Girl | pop | play |
| Gonzalo Rubalcaba | The Hard One | jazz | play |
| Stevie Ray Vaughan | 08 - Little Wing | blues | play |
| Steve Miller Band | The Joker | rock | play |

Q1: How good is this recommended sequence for your group? ****

Q2: How good is this recommended seqence for you personally? ✓ * ** *** **** *****

Q3: To what extend does this recommended sequence contain interesting and unexpected tracks that you think you group would like?

Q4: How good is this recommended sequence compared to the one that you have suggested? similar

show your recommendation sequence

## Recommendation 2:

| Artist | Title | Genre | |
|---|---|---|---|
| Muddy Waters | Long Distance Call | blues | play |
| Nirvana | Come As You Are | rock | play |
| Police | Message In A Bottle | rock | play |
| Police | Roxanne | rock | play |
| Eagles | Hotel California | rock | play |
| Gorillaz | Feel Good, Inc | pop | play |
| The Blues Label | Leadbelly - Pig meat papa | blues | play |
| The Dave Brubeck Quartet | Three To Get Ready | jazz | play |
| Pink Floyd | Hey You | rock | play |
| Pink Floyd | Wish You Were Here | rock | play |

Q1: How good is this recommended sequence for your group?

Q2: How good is this recommended seqence for you personally?

Q3: To what extend does this recommended sequence contain interesting and unexpected tracks that you think you group would like?

Q4: How good is this recommended sequence compared to the one that you have suggested?

show your recommendation sequence

Q5: Which recommended sequence would you select for your group? ◯ sequence 1

# Recommendation Techniques

- **User built:** each group member builds a recommended compilation for his group
- **Averaging:** the tracks with the largest average predicted (or actual) ratings are selected
- **Balancing:**
  - the compilation is generated incrementally
  - at each step a new track is added: that one minimizing the **differences of the accumulated satisfactions of the users**
- **Balancing with decay:**
  - Similar to balancing but in the computation of the user satisfaction at one step the older tracks count less.

# Balancing

- If $S$ is a sequence of tracks and $M$ is the sequence of tracks of equal length with the highest ratings (either predicted or actual) then the satisfaction of $u$ for $S$ is:

$$sat(u,S) = \frac{\sum_{i \in S} r^*(u,i)}{\sum_{j \in M} r^*(u,j)}$$

- If $S_{+i}$ is the sequence extending $S$ with track $i$ then the item added to $S$ by the Balancing rule is such that

$$\underset{i}{Arg\min} \sum_{u,v \in g} \left| sat(u,S_{+i}) - sat(v,S_{+i}) \right|$$

# Balancing Example

| | Track1 | Track2 | Track3 | Track4 | Track5 | Track6 |
|---|---|---|---|---|---|---|
| John | 3 | 2 | 5 | 4 | 5 | 2 |
| Peter | 4 | 5 | 2 | 2 | 1 | 4 |
| Ann | 5 | 4 | 3 | 3 | 4 | 5 |
| Group average: | 4 | 3.67 | 3.33 | 3 | 3.33 | 3.67 |

| | Track1 | Track2 | Track3 | Track5 | Track6 |
|---|---|---|---|---|---|
| John | 3 | 2 | 5 | 5 | 2 |
| Peter | 4 | 5 | 2 | 1 | 4 |
| Ann | 5 | 4 | 3 | 4 | 5 |
| Group average: | 4 | 3.67 | 3.33 | 3.33 | 3.67 |

**Candidate set:** contains tracks with large average predicted ratings

# Balancing Example

Candidate set:

| | Track1 | Track2 | Track3 | Track5 | Track6 |
|---|---|---|---|---|---|
| **John** | **3** | 2 | 5 | 5 | 2 |
| **Peter** | **4** | 5 | 2 | 1 | 4 |
| **Ann** | **5** | 4 | 3 | 4 | 5 |
| **Group average:** | **4** | 3.67 | 3.33 | 3.33 | 3.67 |

Sequence: track1 is the best initial option because has the largest average rating.

# Balancing Example

| | Track1 | Track2 | Track3 | Track5 | Track6 |
|---|---|---|---|---|---|
| **John** | 3 | 2 | 5 | 5 | 2 |
| **Peter** | 4 | 5 | 2 | 1 | 4 |
| **Ann** | 5 | 4 | 3 | 4 | 5 |
| **Group average:** | 4 | 3.67 | 3.33 | 3.33 | 3.67 |

<Track1, Track3> minimizes the satisfaction differences among group members

| Sequence | Sat(John,s) | Sat(Peter,s) | Sat(Ann,s) | Sat differences |
|---|---|---|---|---|
| Track1, Track2 | 5/10 | 9/9 | 9/10 | 1 |
| Track1, Track3 | 8/10 | 6/9 | 8/10 | 0.267 |
| Track1, Track5 | 8/10 | 5/9 | 9/10 | 0.689 |
| Track1, Track6 | 5/10 | 8/9 | 10/10 | 0.999 |

# Balancing Example

| | Track1 | Track2 | Track3 | Track5 | Track6 |
|---|---|---|---|---|---|
| **John** | 3 | 2 | 5 | 5 | 2 |
| **Peter** | 4 | 5 | 2 | 1 | 4 |
| **Ann** | 5 | 4 | 3 | 4 | 5 |
| **Group average:** | 4 | 3.67 | 3.33 | 3.33 | 3.67 |

<Track1, Track3, Track2> is the balancing sequence with 3 tracks

| Sequence | Sat(John,s) | Sat(Peter,s) | Sat(Ann,s) | Sat differences |
|---|---|---|---|---|
| Track1, Track3, Track2 | 10/13 | 11/13 | 12/14 | 0.176 |
| Track1, Track3, Track5 | 13/13 | 7/13 | 12/14 | 0.539 |
| Track1, Track3, Track6 | 10/13 | 10/13 | 13/14 | 0.318 |

# Comparison

- Rank aggregation with average:

| | Track1 | Track2 | Track3 | Track4 | Track5 | Track6 |
|---|---|---|---|---|---|---|
| John | 3 | 2 | 5 | 4 | 5 | 2 |
| Peter | 4 | 5 | 2 | 2 | 1 | 4 |
| Ann | 5 | 4 | 3 | 3 | 4 | 5 |
| Group average: | 4 | 3.67 | 3.33 | 3 | 3.33 | 3.67 |

- Balancing:

| | Track1 | Track2 | Track3 | Track4 | Track5 | Track6 |
|---|---|---|---|---|---|---|
| John | 3 | 2 | 5 | 4 | 5 | 2 |
| Peter | 4 | 5 | 2 | 2 | 1 | 4 |
| Ann | 5 | 4 | 3 | 3 | 4 | 5 |
| Group average: | 4 | 3.67 | 3.33 | 3 | 3.33 | 3.67 |

# Experimental setup

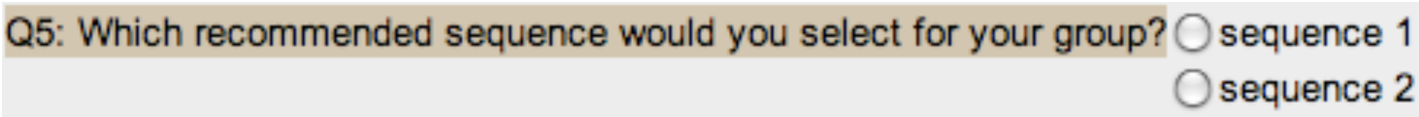
- Large scale live user study
- Fully functional sequential group recommender
- We compared:
  - ‘Balancing’ without Decay
  - ‘Balancing’ with Decay
  - Average
  - User generated
- Participant tasks included:
  - Rate music tracks
  - Get assigned into groups
  - Compile a sequence suggestion to one’s group
  - Evaluate other track sequences

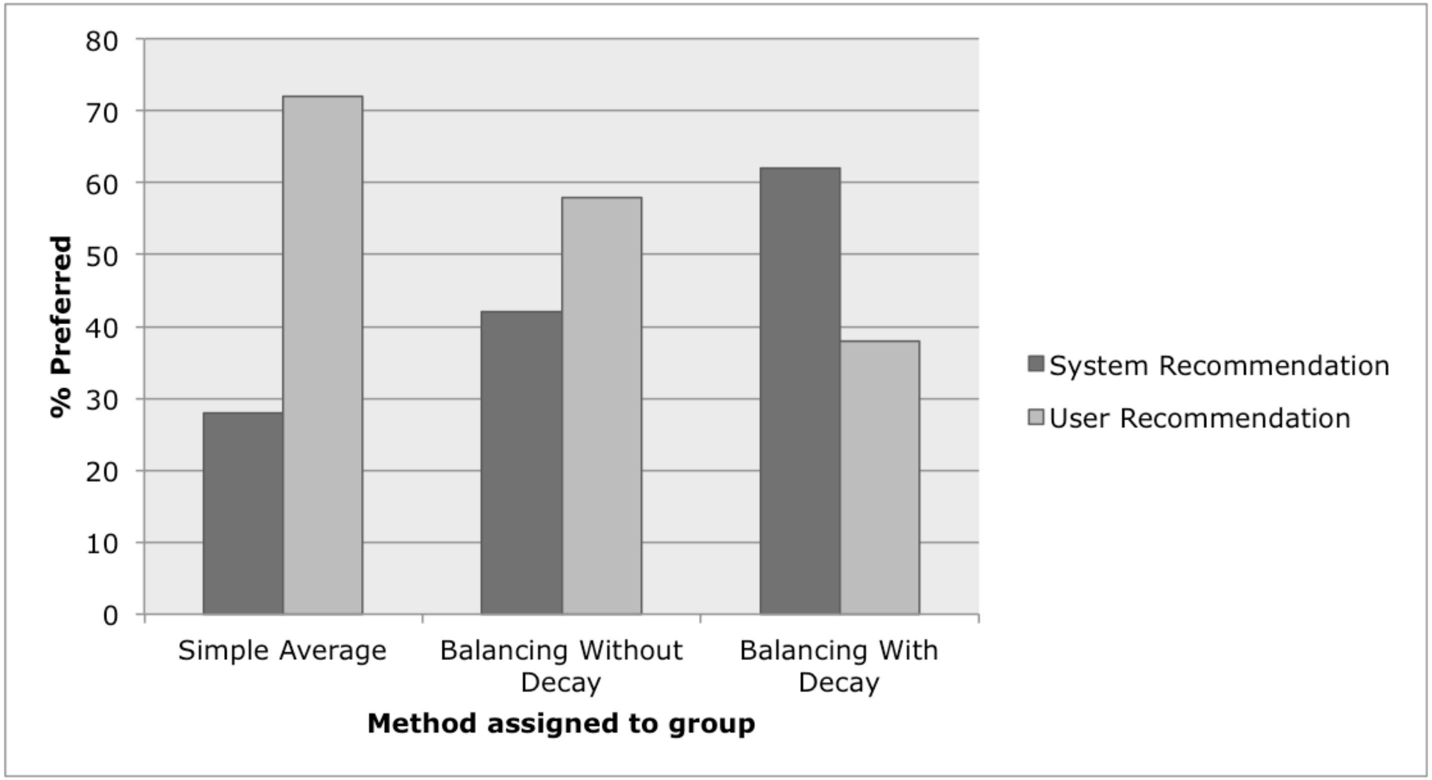
# Experimental setup II

- Music track corpus of 1068 tracks
- 77 users have left 5160 ratings with the average of 67 ratings per user and 5 ratings per track
- Out of 38 groups created 32 have finished the experiment at least partly
- Each group was assigned one of the three methods to be tested: ‘Average’, ‘Balancing without Decay’ and ‘Balancing with Decay’

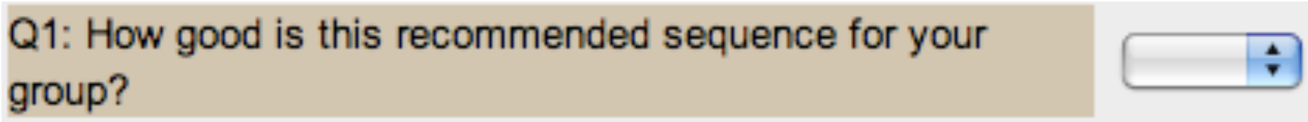
# Results: preferred sequence

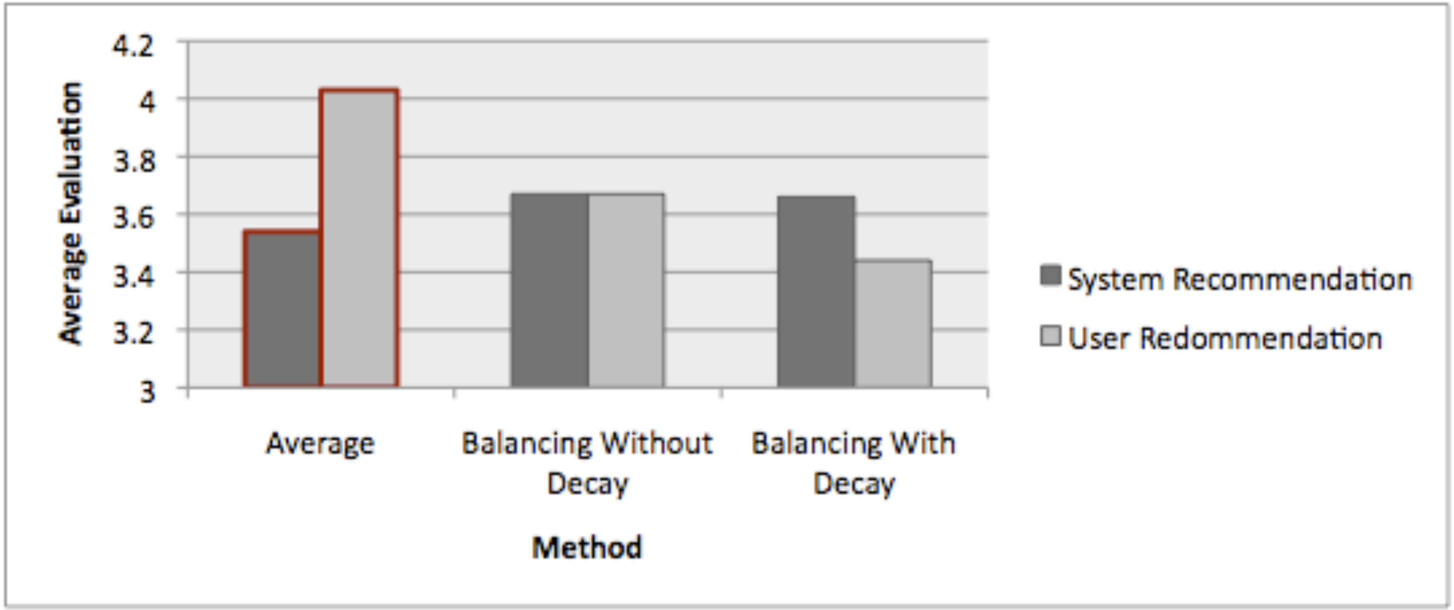
- Choice between system produced and human made recommendations:



# Results: goodness for group

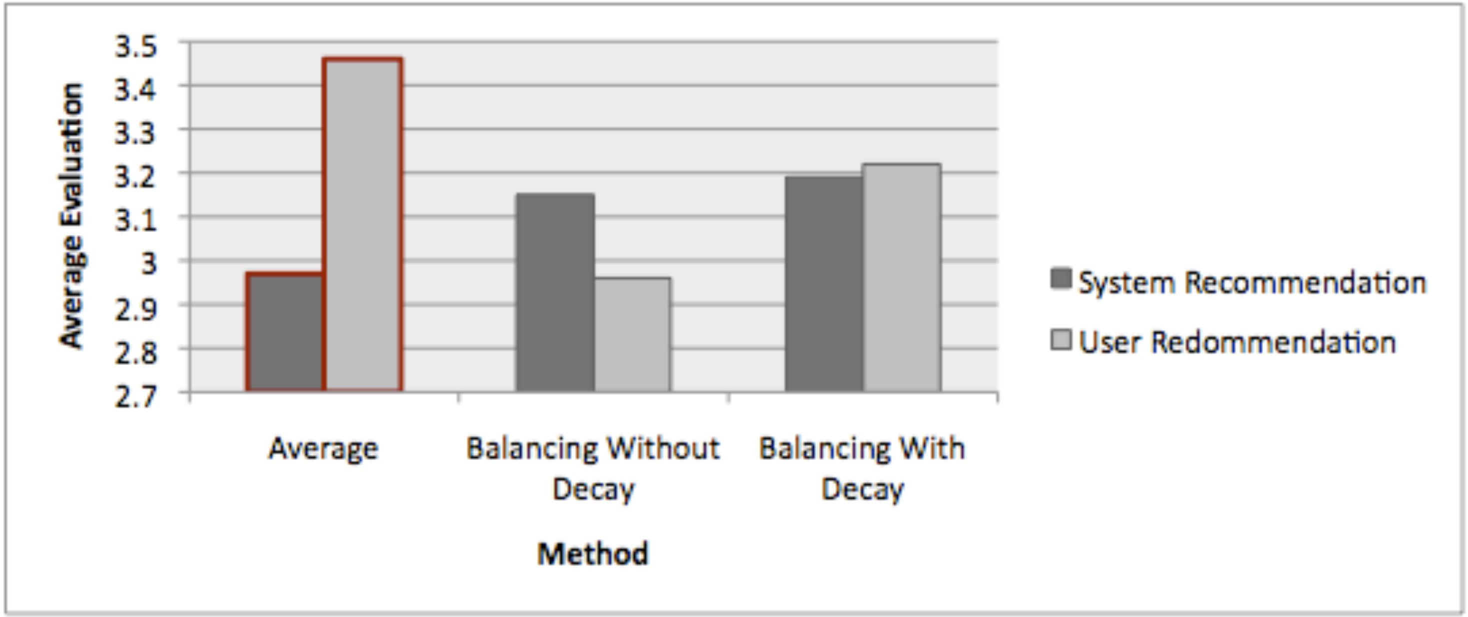



#of users per condition: Average 39; Balancing 26, Balancing with decay 24.

# Results: novelty

Q3: To what extend does this recommended sequence contain interesting and unexpected tracks that you think you group would like?



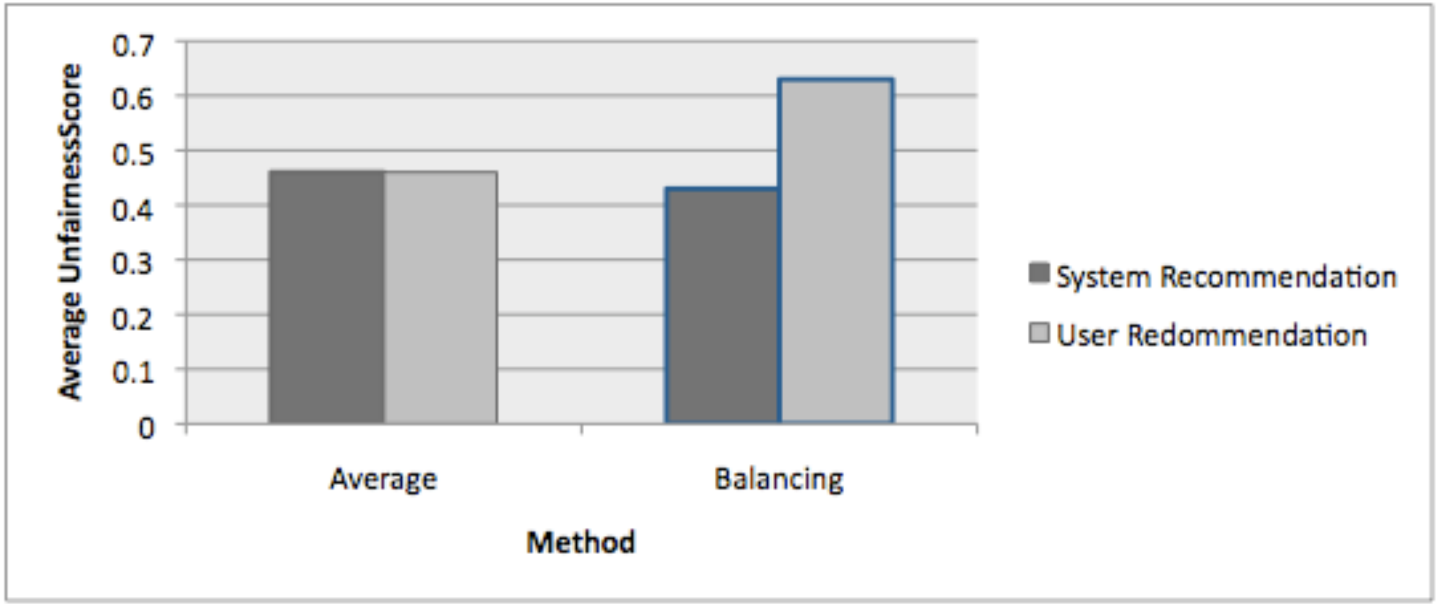
# Results: fairness

- Group recommendation is fair if the following two are close:
  - Goodness for group (Q1)
  - Personal satisfaction (Q2)

Q1: How good is this recommended sequence for your group?

Q2: How good is this recommended seqence for you personally?

- For each group member calculate the absolute difference: |Q1 - Q2|
- Take an average of those differences as an unfairness score for the group (the smaller the score, the better results)



# Human Rec. Strategies

- More than 10 strategies were found analysing the user comments about how they built music track sequences

| Strategy type | Comment |
|---|---|
| **Intersection of everyone's preferences** | "Sorted tracks by user evaluations and picked the ones that all group members marked with 5 or 4 stars." |
| **Compromise (a bit for each)** | "Chose songs, highly rated by one of the members, each member a few." |
| **Compromise (at least not hated by anybody)** | "Not many ratings in common... so I chose songs which had minimum 3 stars from minimum 2 users." |
| **Guessing/reasoning from available information** | "First, I looked for tracks with high ratings by all members. I then filled up the list with tracks that were rated by one member only but, based on what other members liked, I thought they would have been rated highly by the other members as well, had they listened to them."; |
| **Own preferences first** | "tracks I like and which have some more stars than other ones at least for one other group members" |
| **Egoistic** | "I have chosen the baroque style music, since it is not very popular among people, but I think everyone should be at least familiar to it."; |

# Conclusions (I)

- Rank aggregation techniques provide a viable approach to group recommendation
- Group recommendations may be better than individual recommendations
  - Both for random groups and high similar groups
- *Users are more similar among them as one can expect*
- It could be used as an individual recommendation technique: search for similar users – make individual predictions to all of them and then aggregate the predictions for the target user *(under further investigation)*
- Groups with high inner similarity (generally) have better group recommendations.

# Conclusions (II)

- First online study where users evaluated system generated group recommendations (vs. user generated)
- For generating sequences of recommendation ‘Balancing’ outperforms state of the art (averaging)
- Balancing performs well even compared to human-made recommendations
- ‘Average’ method inferior to human recommendations when considering:
  - Overall quality
  - Goodness for the group
  - Novelty

# References


- Arrow, K.J. (1970) Social Choice and Individual Values. Yale University Press, second edition, 1970.
- Baccigalupo, C. Poolcasting– An Intelligent Technique to Customise Music Programmes for Their Audience. PhD Thesis, UAB, 2009.
- Baltrunas, L., Kaminskas, M., Ludwig, M., Moling, O., Ricci, F., Aydin, A., Lueke, K. and Schwaiger, R. InCarMusic: Context-Aware Music Recommendations in a Car. 12th International Conference on Electronic Commerce and Web Technologies - EC-Web 201, Toulouse, France, pages 89-100, 2011.
- Baltrunas, L., Makcinskas, T., Ricci, F. Group recommendations with rank aggregation and collaborative filtering. In: RecSys 2010: Proceedings of the 2010 ACM, Conference on Recommender Systems, pages 119–126, 2010.
- Celma, O. and Lamere, P. If you like Radiohead, you might like this article. *AI Magazine*, volume 32, number 3, pages 57–66, 2011.

# References


- Dwork, C., Kumar, R., Naor, M. and Sivakumar, D. Rank aggregation methods for the Web. *Proceedings of the 10th international conference on World Wide Web (WWW '01)*, New York, NY, USA, pages 613-622, 2001. ACM.
- Fields, B. Contextualize Your Listening: The Playlist as Recommendation Engine, PhD Thesis, Goldsmiths, University of London, April 2011.
- Jameson, A. More than the sum of its members: challenges for group recommender systems. Proceedings of the working conference on Advanced visual interfaces (AVI '04). ACM, New York, NY, USA, pages 48-54, 2004.
- Jameson, A. and Smyth, B. Recommendation to groups. In P. Brusilovsky, A. Kobsa, and W. Nejdl, (eds.), The Adaptive Web, volume 4321 of Lecture Notes in Computer Science, pages 596–627, Springer, 2007.
- Kemeny, J. (1959) Mathematics without numbers. *Daedalus*, volume 88, pages 577-591, 1959.

# References


- Masthoff, J. Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers. UMUAI, volume 14, pages 37-85, 2004.
- Masthoff, J. and Gatt, A. In pursuit of satisfaction and the prevention of embarrassment: affective state in group recommender systems. User Modeling User-Adapted Interaction, volume 16, issue 3-4, pages 281–319, 2006.
- Masthoff, J. Group recommender systems: Combining individual models. In Ricci, F., Rokach, L., Shapira, B., Kantor, P. (Eds.), Recommender Systems Handbook (pp. 677-702). Springer-Verlag, 2011.
- Piliponyte, Auste. Sequential Group Recommendations. MA Thesis, Free University of Bozen–Bolzano, 2012.

# Questions?