Recommendation Based on Personal-values: beyond Recommending What You Might Prefer

YASUFUMI TAKAMA
TOKYO METROPOLITAN UNIVERSITY, JAPAN

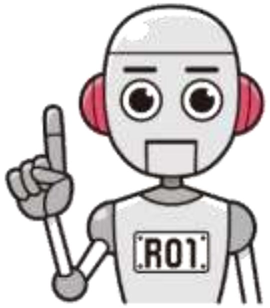
Research Topics

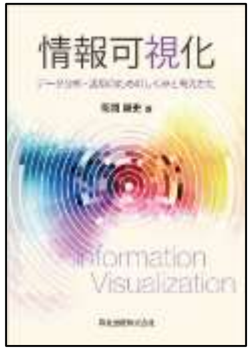
Web Intelligence

Recommendation

Support for improving
AI performance

Human in the loop





InfoVis.

Support for information
access/understanding

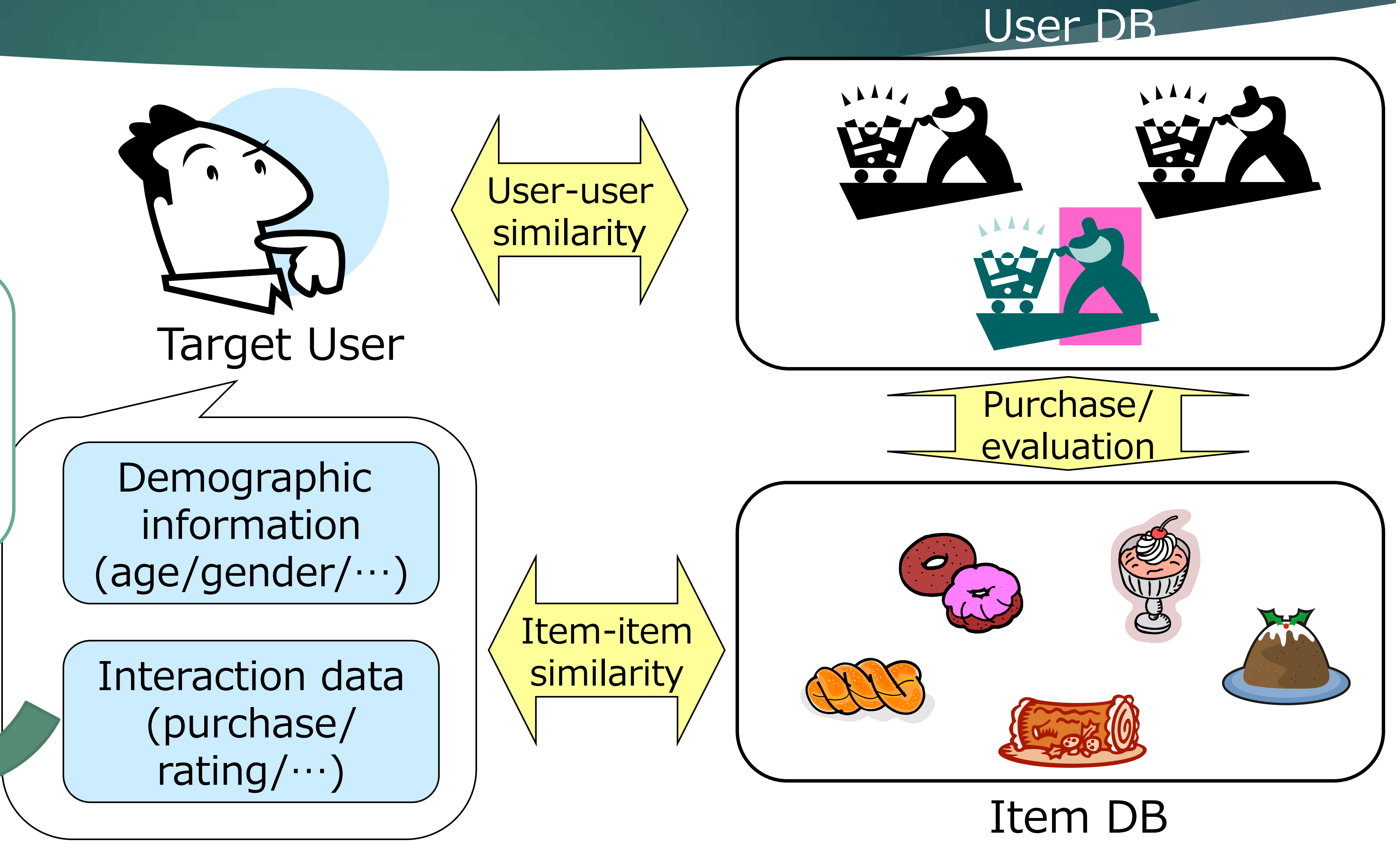
Table of Contents

- Introduction to Information Recommendation
  - Aim of recommendation
  - Algorithms
  - Evaluation metrics
- Beyond Accuracy
  - Challenges
  - Personal values
  - Introduction to collaborative filtering
  - Extension: user modeling from browsing history, item modeling

Recommendation is …

- Find items of interest to target user from vast amount of items

Used Information for Recommendation



- Binary data (implicit)
  - Purchase = like
- Ordinal data (explicit)
  - 5-scale rating

Assumption behind Recommendation

- Similar users have similar preference for items
  - **Purchased same items in the past**
  - Similar demographic information

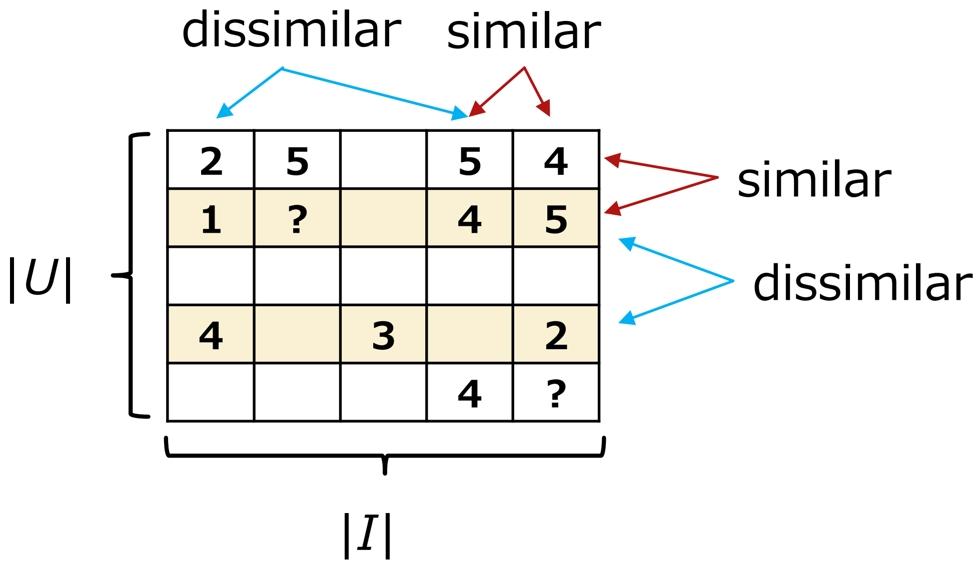
  Conditions of similar users

- Users prefer items similar to those they preferred in the past
  - Movies of same categories
  - New album of favorite singer

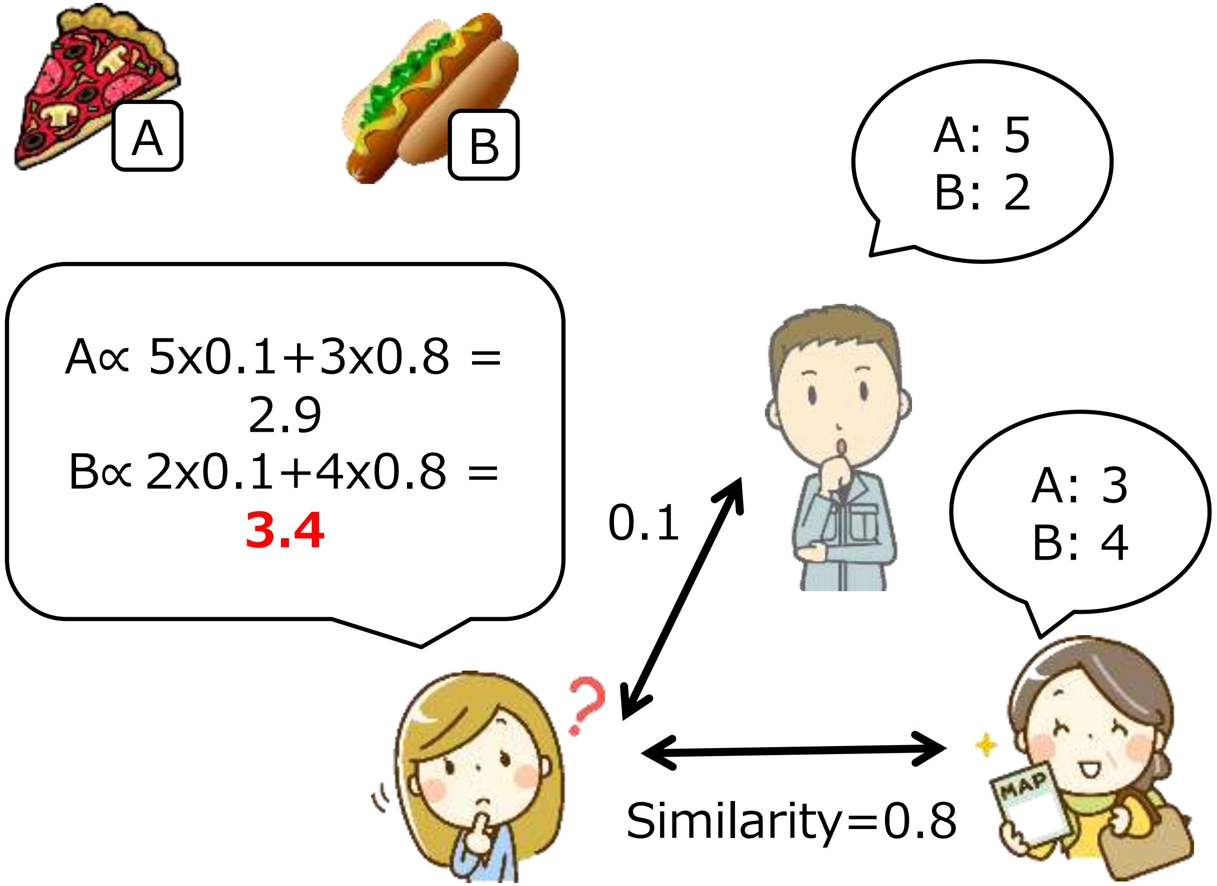
Collaborative Filtering

Collaborative Filtering

- Rating Matrix
  - Record of user-item interaction
  - Value
    - Rating ··· 1:bad – 5:good
    - Implicit feedback ··· 1:buy – 0:not yet
  - Predict unknown rating value
- Neighborhood-based approach
  - User-based: similar user = similar ratings to same items
  - Item-based: similar item = similar ratings by same user



Neighborhood-based CF



Rating matrix

4	x	2	1	x	x
4	x	x	x	4	x
x	3	x	2	2	x
x	x	2	5	x	4
x	x	1	4	3	4

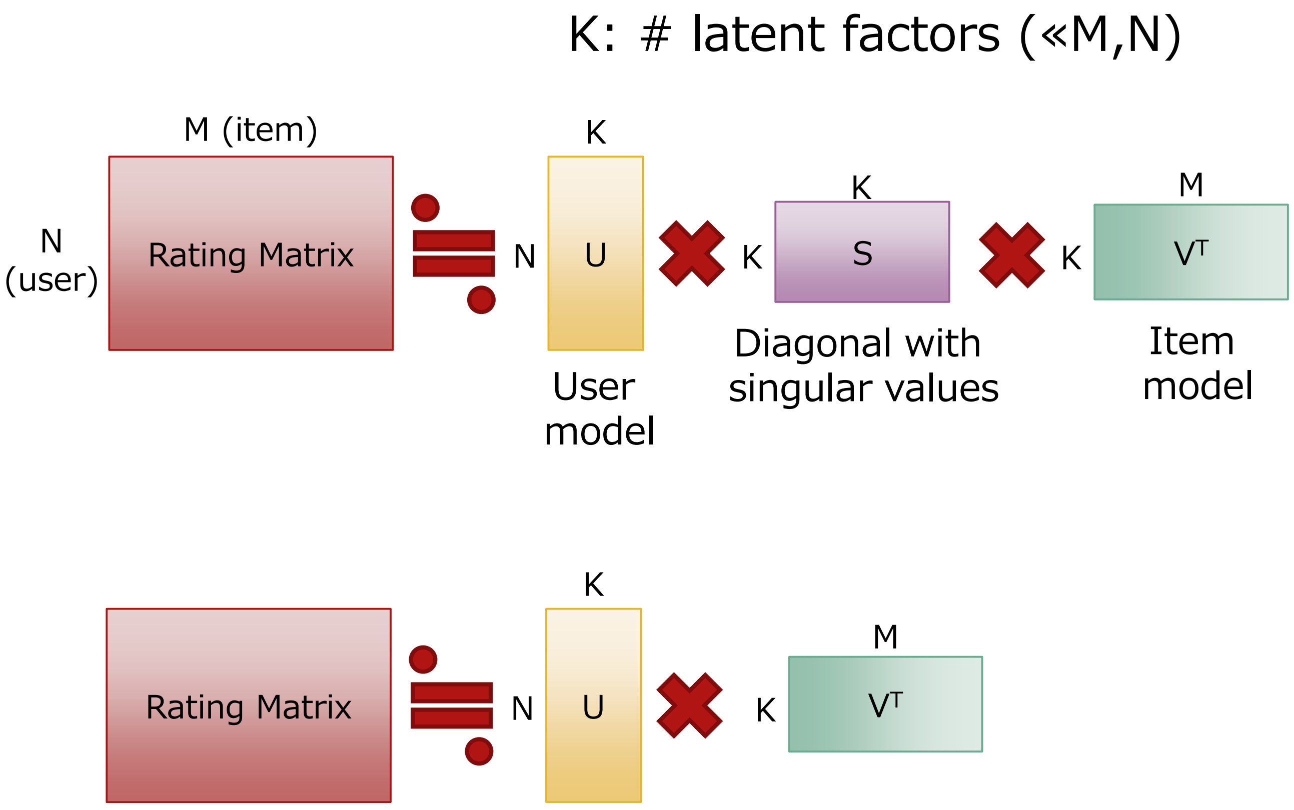
Similarity between vectors

- Prediction by weighted average
  - Rating × similarity
- Similarity of user vectors
  - Cosine
  - Pearson correlation coefficient

Matrix Factorization-based CF

- Neighborhood-based CF = Memory-based approach
  - User/item vector = row/column of rating matrix
  - Too sparse: few common items rated by different users
  - Cold start problem, sparsity problem
- Solution: dimensionality reduction
  - Rating matrix ⇒ user models, item models with lower dimensions
  - Prediction by dot product of item/user vectors
  - Model-based approach

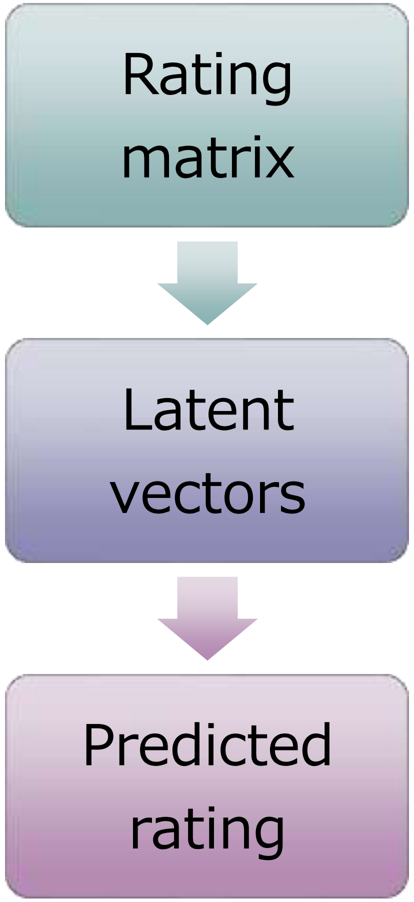
Variations of Matrix Factorization-based CF



- SVD (Singular Value Decomposition) [Sarwar00]
- NMF (Non-negative Matrix Factorization) [Lee00]
  - U, V: non-negative values
- PMF (Probabilistic Matrix Factorization) [Salakhutdinov07]
  - Rating $\sim N(UV^T, \sigma^2)$

Model-based CF

- Matrix Factorization-based CF (MCF)
- Neural-based CF (NCF)[He17]
- Common strategy
  - Learning latent factors for user/item
- Difference in predicted rating calculation
  - MCF: linear function ⋯ dot product
  - NCF: nonlinear function



Evaluation Metrics

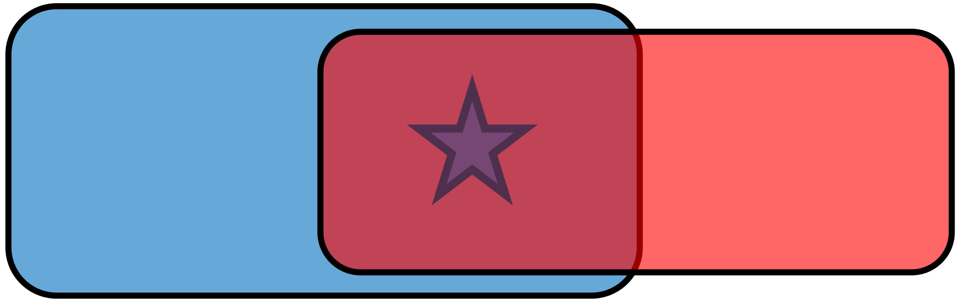
- Prediction error
  - MAE (Mean Absolute Error)
  - RMSE (Root Mean Square Error)
- Top-N recommendation
  - Precision: ★÷■
  - Recall : ★÷■

Actual rating	**5**	**3**	**2**
Predicted rating	4	3	4

$$MAE = \frac{|5-4| + |3-3| + |2-4|}{3} = 1.0$$

$$RMSE = \sqrt{\frac{(5-4)^2 + (3-3)^2 + (2-4)^2}{3}} = 1.29$$

Recommend
N items



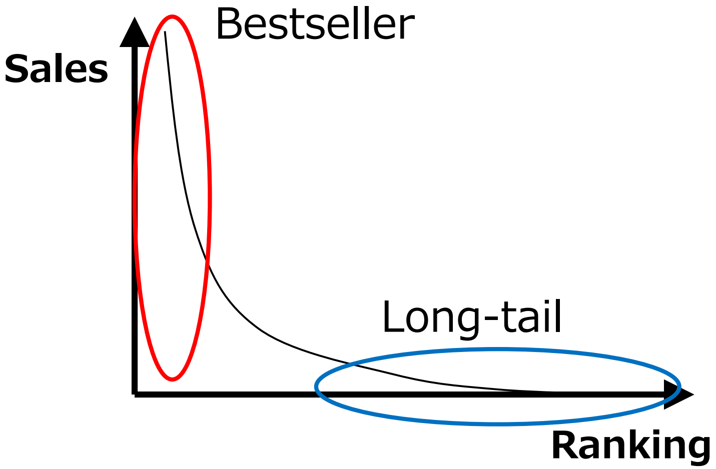
favorite
items

Beyond Accuracy

- Traditional challenge
  - Cold start problem: new users, new items
  - How to achieve high accuracy for new users?
- Recent challenges
  - Context awareness: location, time of day, weekday/weekend, etc.
  - **Long-tail items:** recommend unpopular items
  - **Diversity:** recommend different set of items
  - **Behavior change:** recommend different actions from past

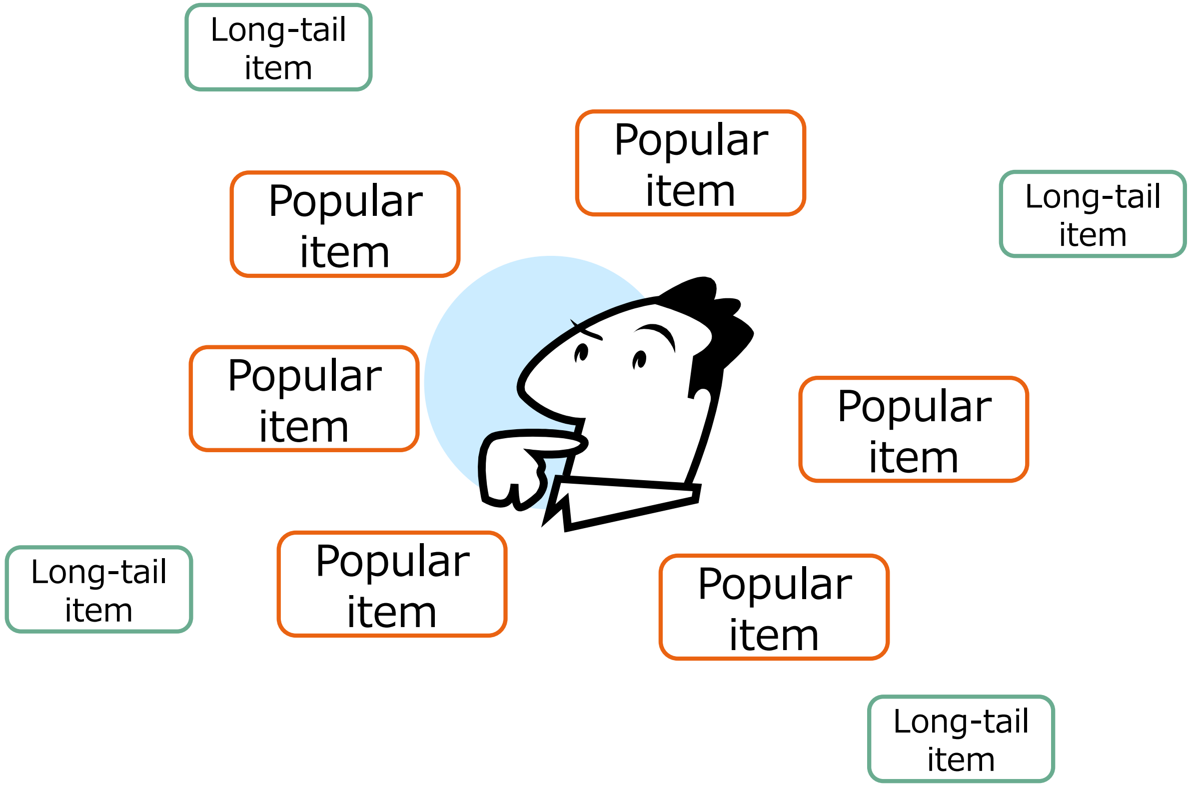
Long-tail Item Recommendation

- Long-tail: unpopular item
  - Amazon: 1/3 of sales from long-tail items (past)
  - Common practice: 80 % of sales from 20% popular items
  - Head area << tail area
  - Difficult in brick & mortar shops
- Merit for seller (company)
  - Gain of sales
- Merit for customers
  - Personalized service ⇒ customer satisfaction ↑



Difficulty in recommending long-tail items

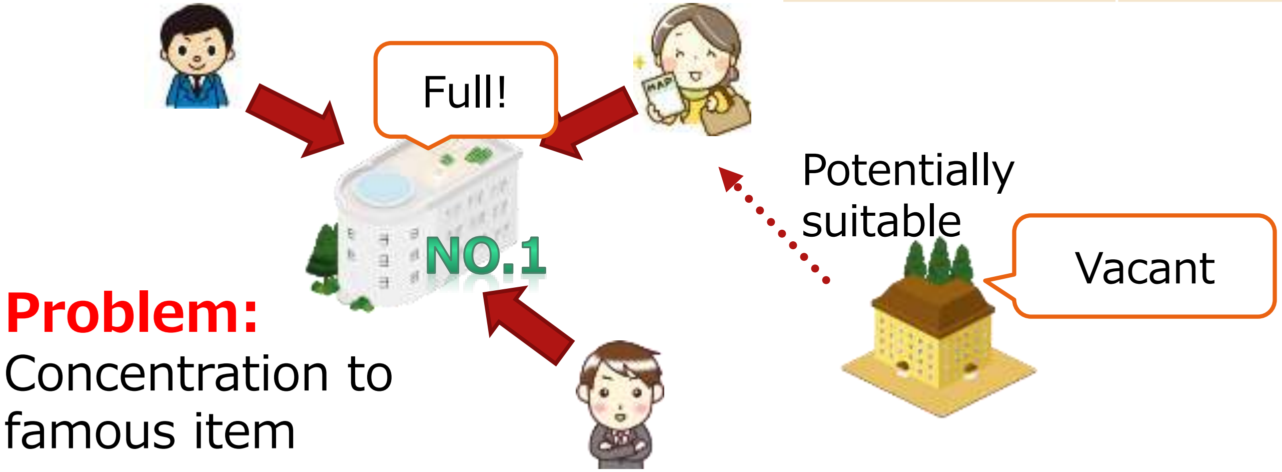
- Popularity bias
  - Popular item:
    - Attract positive ratings
  - Recommend to many users
    - Regardless of CF algorithms
- Solution
  - Consider other factors than accuracy
  - e.g. Diversity



Diversity

homogenous	Diversified
Star Wars EP1	Star Wars EP1
Star Wars EP2	Walking dead
Star Wars EP3	Peter Rabbit
Matrix	KAN-WOO
Solo: SW story	Oceans

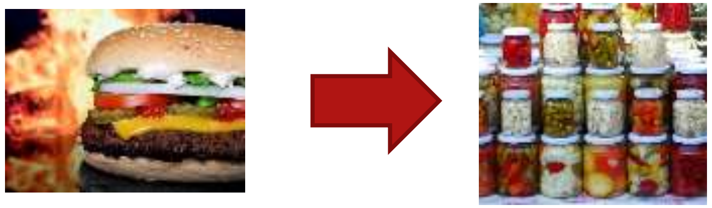
- [Within user] Different types of items for a user
  - Different genres, artists, topics, etc.
- [Between users] Different items for different users
  - Useful for solving social concerns
    - Hotels, restaurants
- **Long-tail items** contribute to diversification



Behavior Change

- Social concern in modern society
  - Health promotion
    - Walking route recommendation
    - Healthy food/recipe recommendation
  - Energy-saving behavior
  - Infection prevention
- Challenges
  - Past behavior is meaningless: Favorite ≠ profitable
  - From Favorite items to profitable & **Acceptable items**
  - **Explanation**: Why this items is recommended

Past **Future**



Frequent light off

Peak-shift

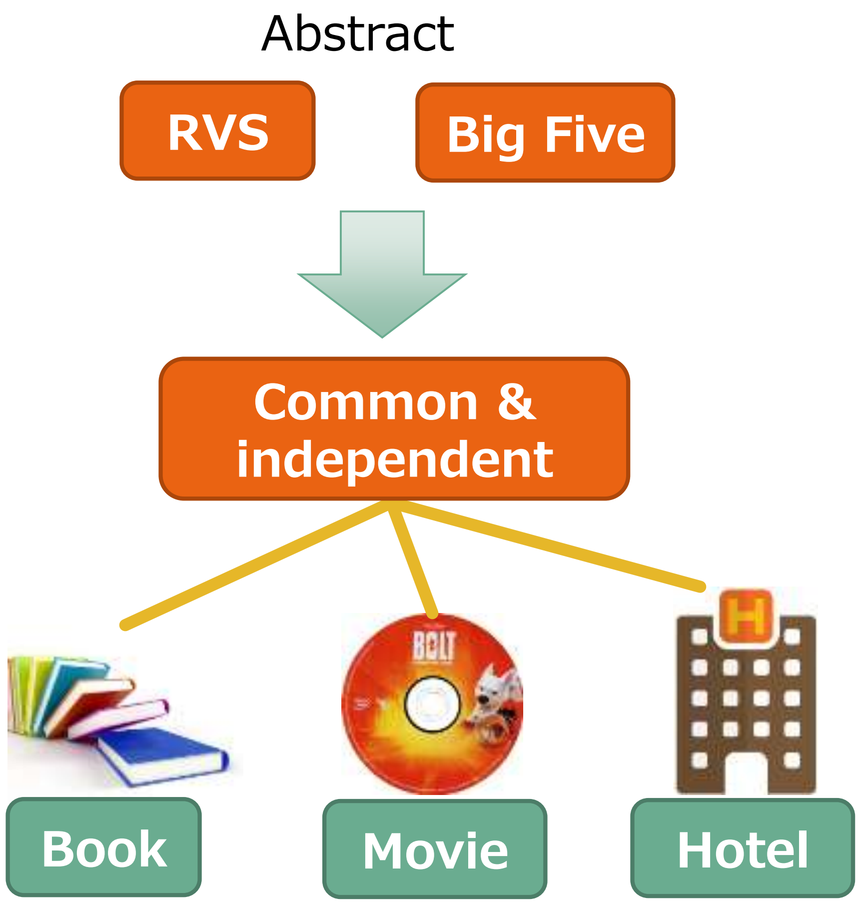
Green Curtain

Personality & Personal Values

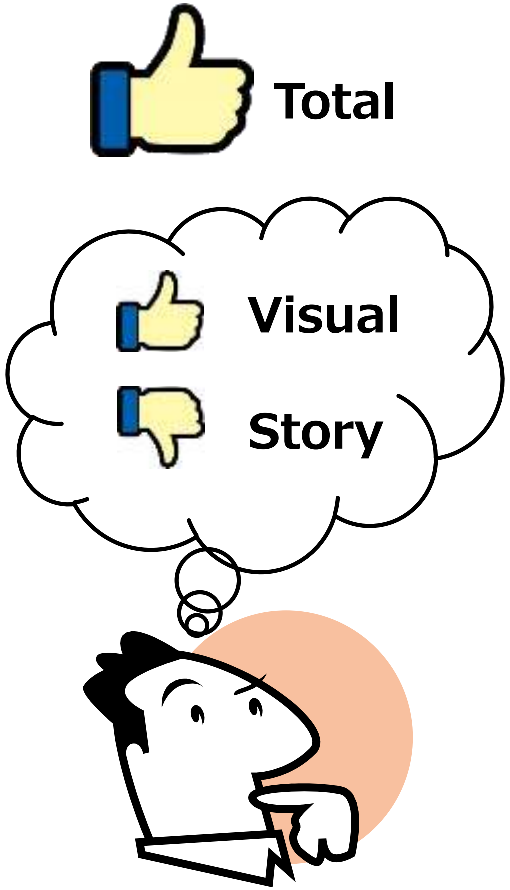
- Personal values
  - Basis for ethical action
  - Acquired nature
  - Rockeach value survey (RVS)
  - Terminal values (18 items)
    - End-states of existence
    - True friendship / Happiness / etc.
  - Instrumental values (18 items)
    - Preferable modes of behavior
    - Ambition / Love / Courage / etc.
- Personality
  - Individual difference among people in behavior patterns, cognition, emotion
  - Inherent nature
  - Big-five factors
    - Openness to experience
    - Conscientiousness
    - Extroversion
    - Agreeableness
    - Neuroticism

Challenge for Personal Values-based Recommendation

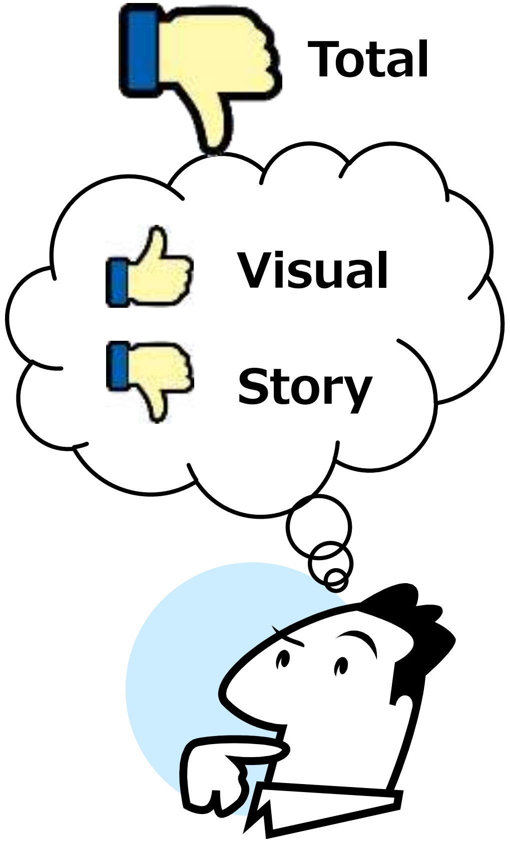
- Distance to preference
  - What to recommend to "***Ambitious***" user?
  - Difficult to directly apply to recommendation
- Independent of target item domain
  - Modeling method should be common to any items
- Possibility of computation
  - Without interpretation / tuning by human expert
  - Implicit modeling



Personal Values as Important Attributes for Decision Making



Movie



Both users agree in *attribute* level

BUT

Total evaluation is different

Different personal values

Rating Matching Rate (RMR)

Review1

Attribute	Polarity
Total	**Positive**
Story	**Positive** ✔
Actor	**Positive** ✔
Music	Negative

Review2

Attribute	Polarity
Total	Negative
Story	Negative ✔
Actor	**Positive**
Music	**Positive**

✔ Same polarity as total evaluation

RMR

Attribute	Story	Actor	Music
Match	2	1	0
Unmatch	0	1	2
RMR	1.0	0.5	0.0

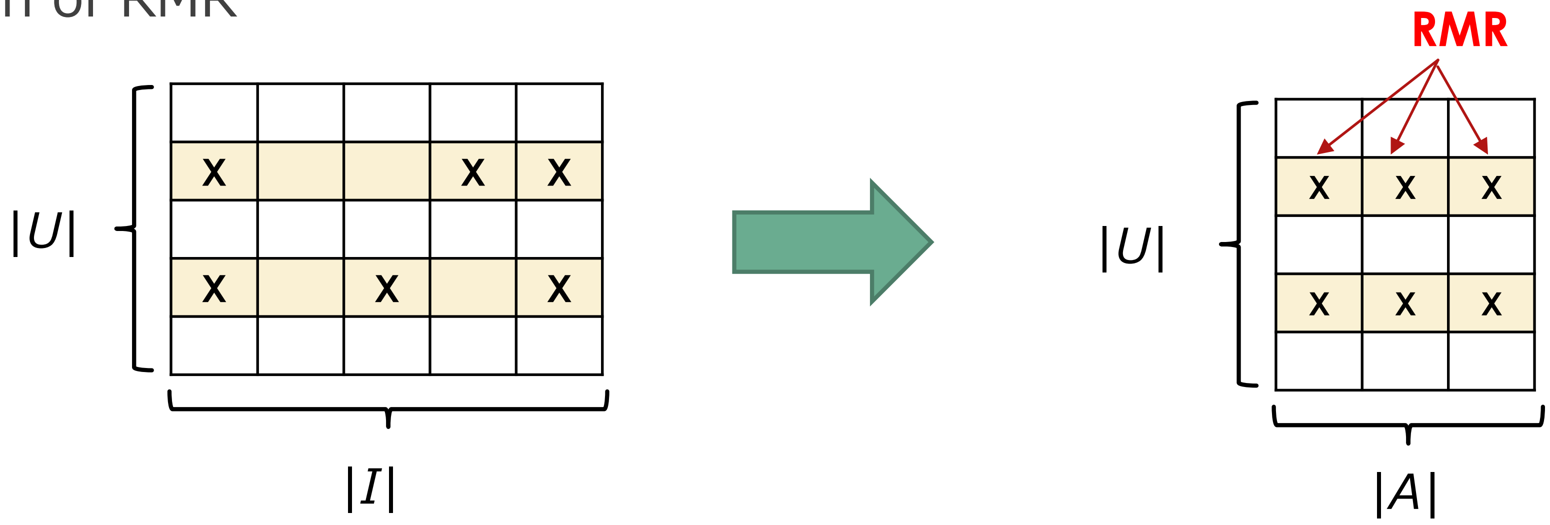
- User model = n-dimensional vector consisting of each attribute's RMR
- High RMR = strong effect on decision making

Advantage of Personal Values-based User Modeling

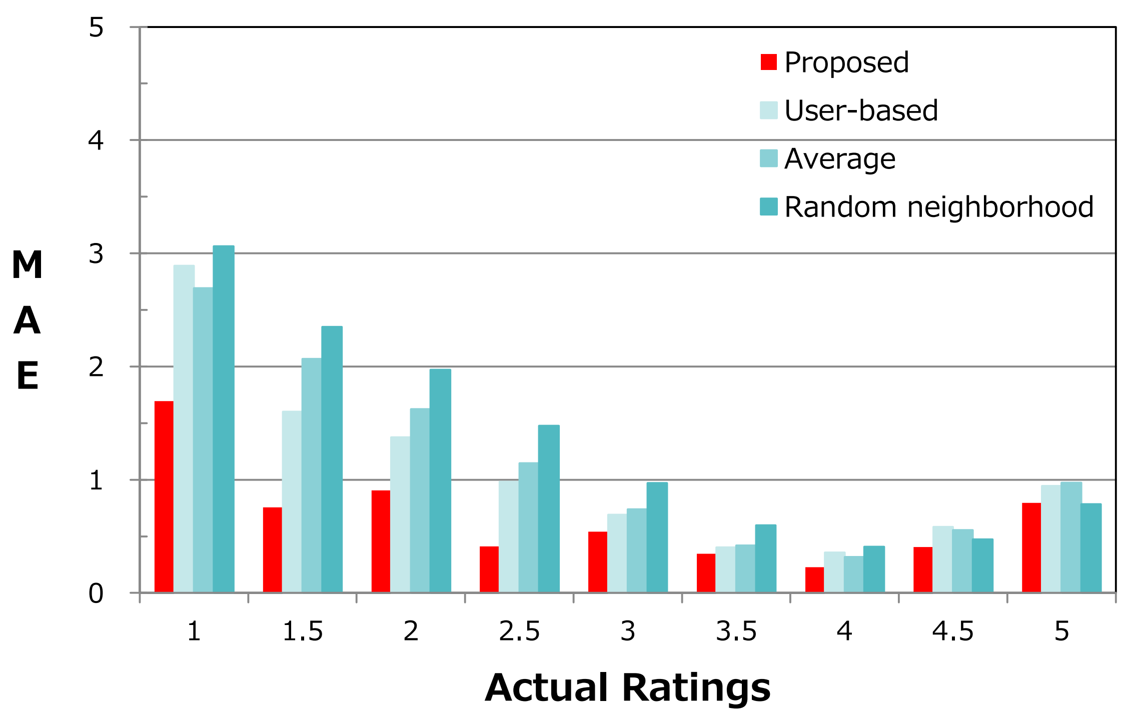
- Model is constructed on attribute space of target item
  - Easy to combine with ordinary recommendation methods
  - Can be calculated for any attribute IF rating is given
- Stable modeling with small number of reviews (<10)
  - Effective for “lack of information” problem
- Potential for
  - Interpretability: suitable for **Explanation** of recommendation
  - Recommending **Acceptable items**: satisfy important attributes
  - Recommending **Long-tail items**: shown by experiments

Personal Values-based Collaborative Filtering (Neighborhood-based CF)

- Extend User-based collaborative filtering
- Used for user-user similarity calculation
  - Baseline: correlation of item ratings (i.e. neighborhood-based)
  - Proposed: correlation of RMR

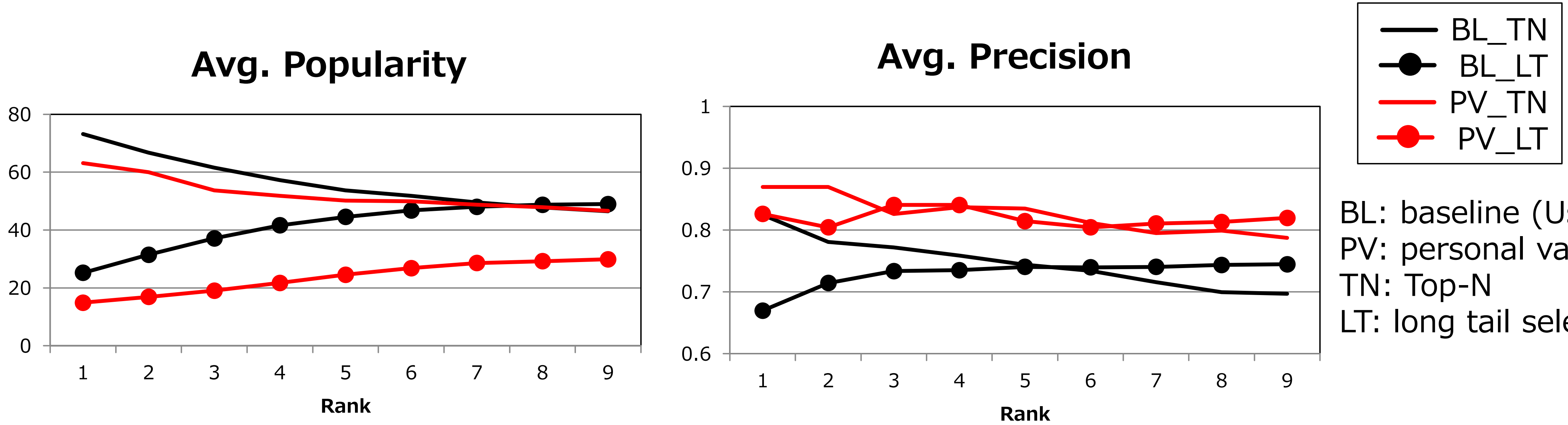


Experimental Result



- Target data: 4Travel
  - 5,079 users
  - 7,295 hotels
  - 64,137 ratings: sparse dataset
- Comparison of MAE
  - All methods achieved lower MAE for around 4
  - Proposed method: lower MAE for lower ratings

Potential for Long-tail item recommendation



BL: baseline (User-based)
PV: personal values
TN: Top-N
LT: long tail selection

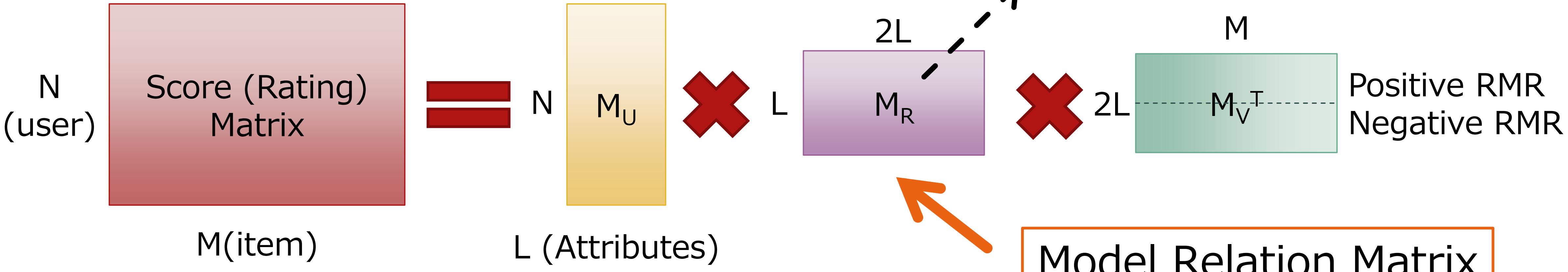
- Long Tail selection: select unpopular items with high predicted ratings
  - PV can enhance effect of Long Tail selection
- PV can improve precision

MCFPV (Matrix-based CF employing Personal Values)

- Difference from usual approach
  - Latent factors ⇒ Item's attributes
  - User / Item models: RMR
  - Recommend higher score items

Interpretability

Large value in M_R(story, cast)
⇒Users care about casts' reputation
if they put priority on story.



Model Relation Matrix

Model Relation Matrix

- Manual Setting[Shiraishi17]
  - Diagonal matrix
- Learning from rating matrix
  - Based on prediction error
  - BPR (Bayesian Personalized Ranking) [Rendle09]

$$\begin{pmatrix} 1 & \cdots & 0 & -1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & -1 \end{pmatrix} \quad \begin{pmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \end{pmatrix}$$

$$\begin{pmatrix} 2 & \cdots & 0 & -1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 2 & 0 & \cdots & -1 \end{pmatrix}$$

$$M_R = \begin{pmatrix} w_{1,1} & \cdots & w_{1,L} & w_{1,L+1} & \cdots & w_{1,2L} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ w_{L,1} & \cdots & w_{L,L} & w_{L,L+1} & \cdots & w_{L,2L} \end{pmatrix}$$

M_U

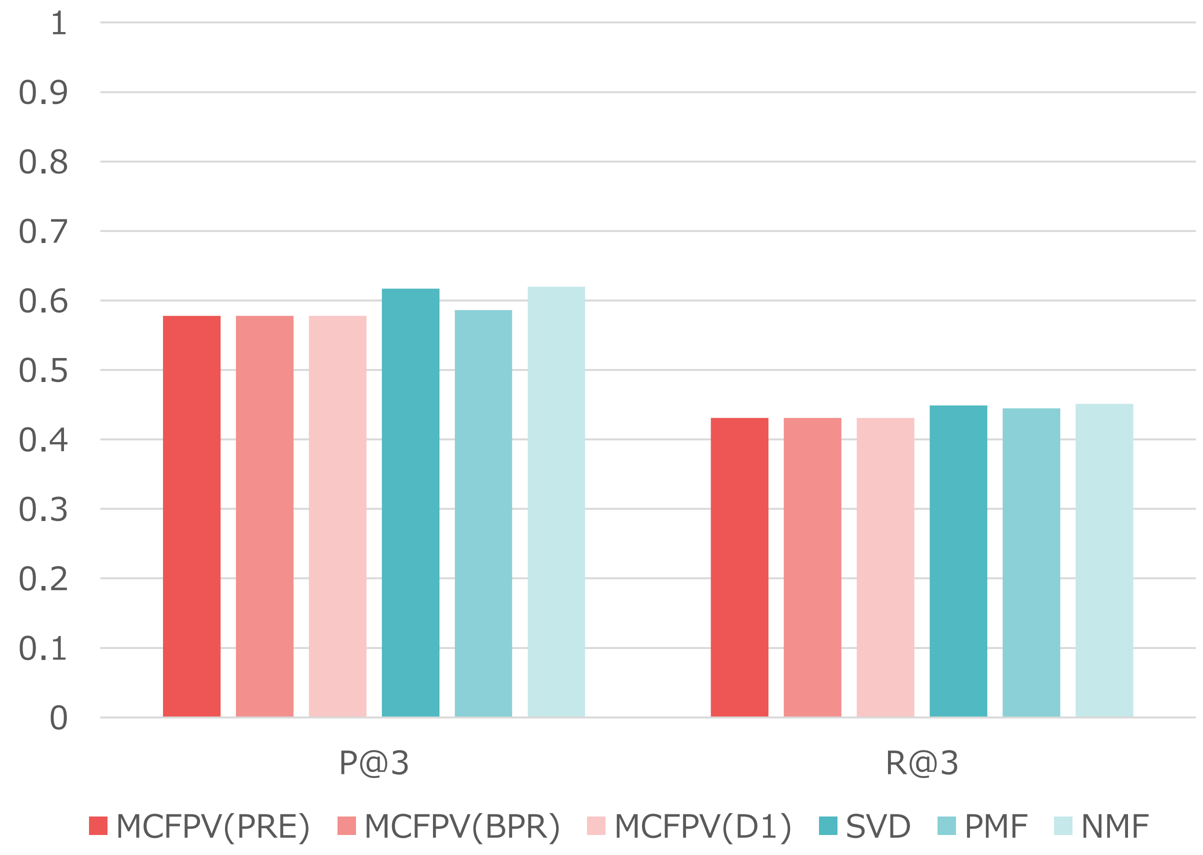
(Positive) M_V (Negative)

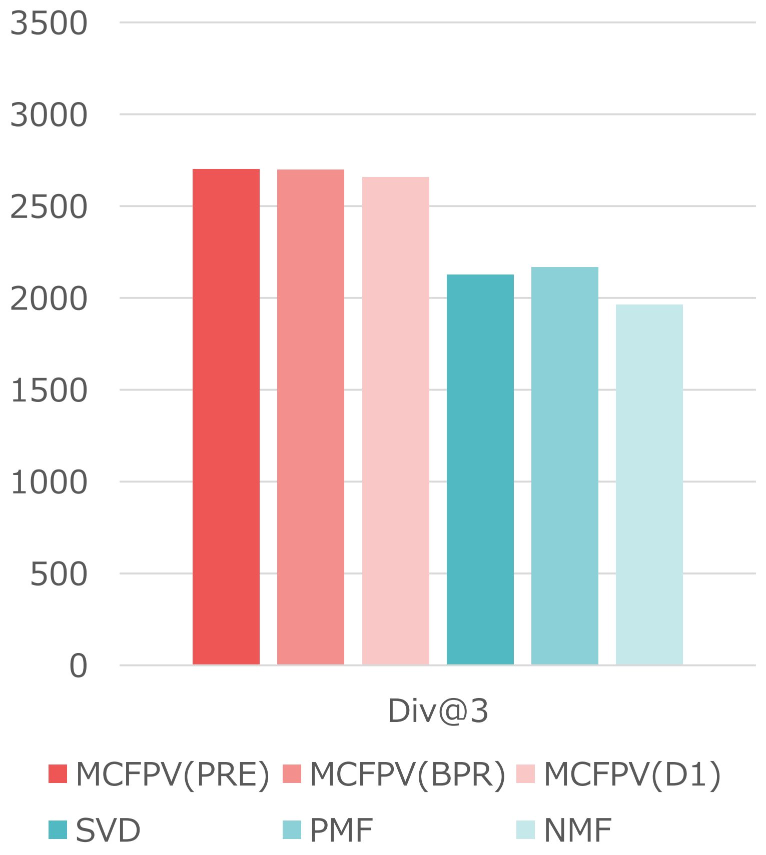
Experiments: Dataset

Dataset	# User	# Item	# Rating	Density
Yahoo! Movie	18,507	6,746	523,730	0.00420
Hotpepper Beauty	31,976	8,101	72,386	0.00028

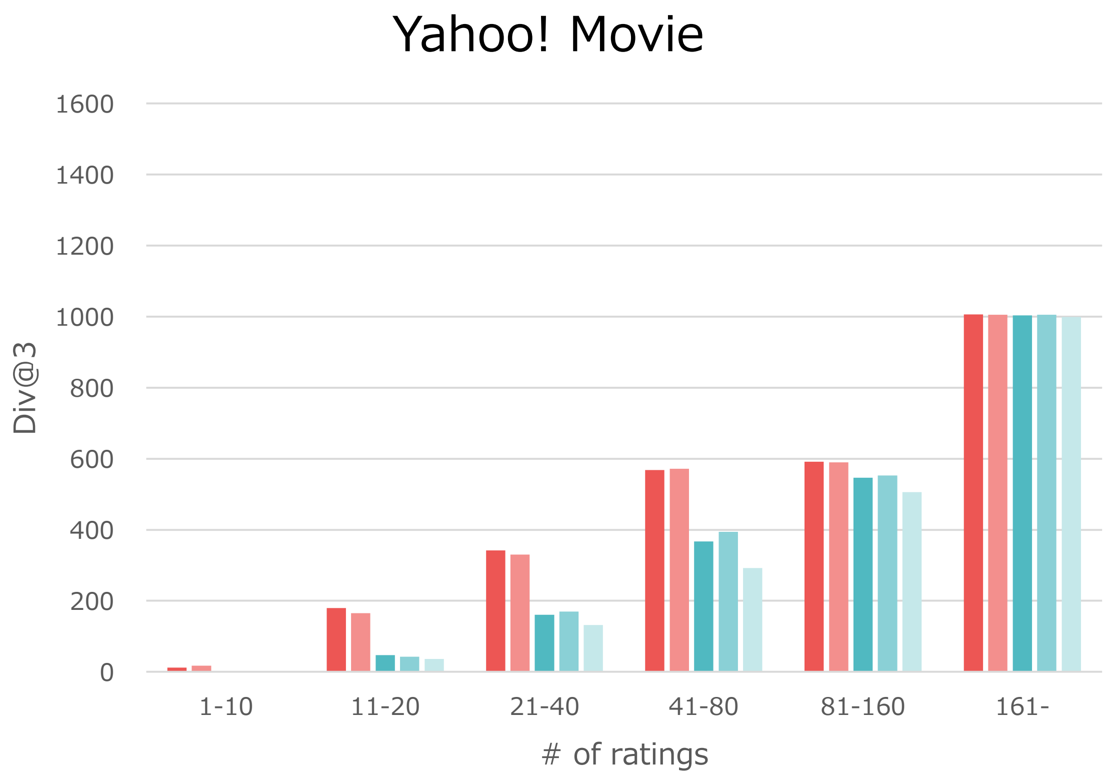
- Yahoo! Movie: rating ∈{1,2,..,5}
  - 5 Attributes: Story, Cast, Scenario, Visuals, Music
- Hotpepper Beauty: rating ∈{1,2,..,5}
  - 4 attributes: Atmosphere, Service, Skill, Price

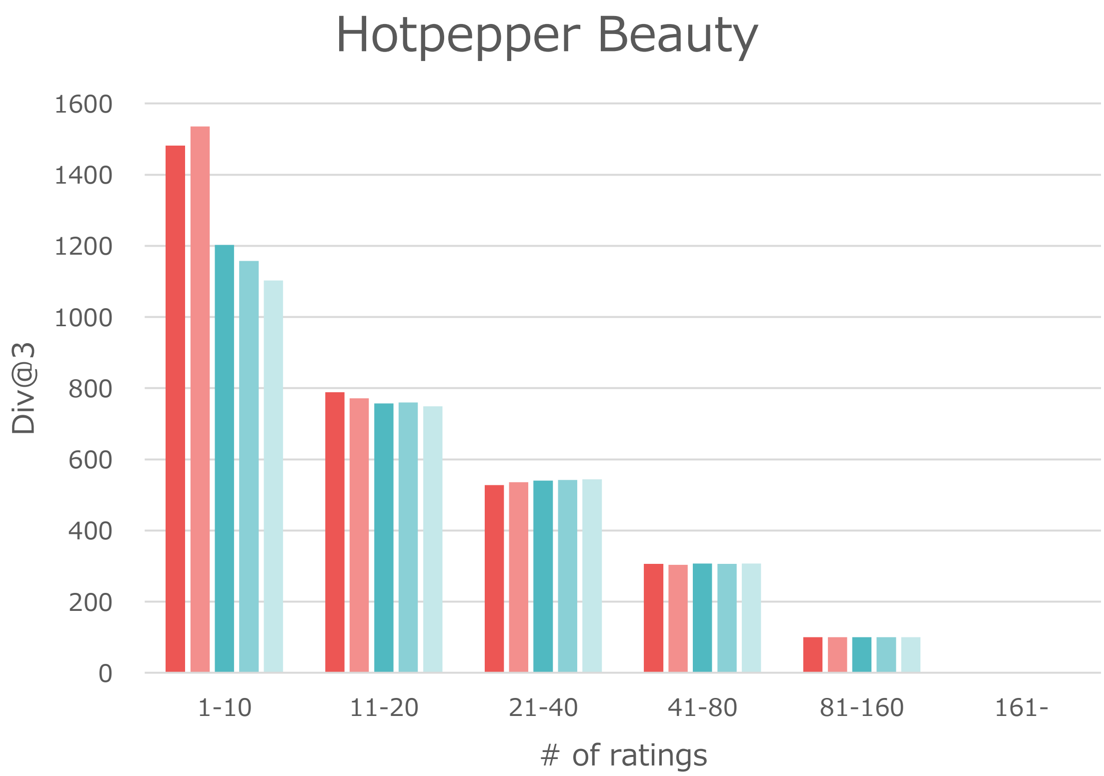
Result: P@3, R@3, Div@3 Yahoo! Movie





Result: (X) Popularity vs. (Y) Diversity



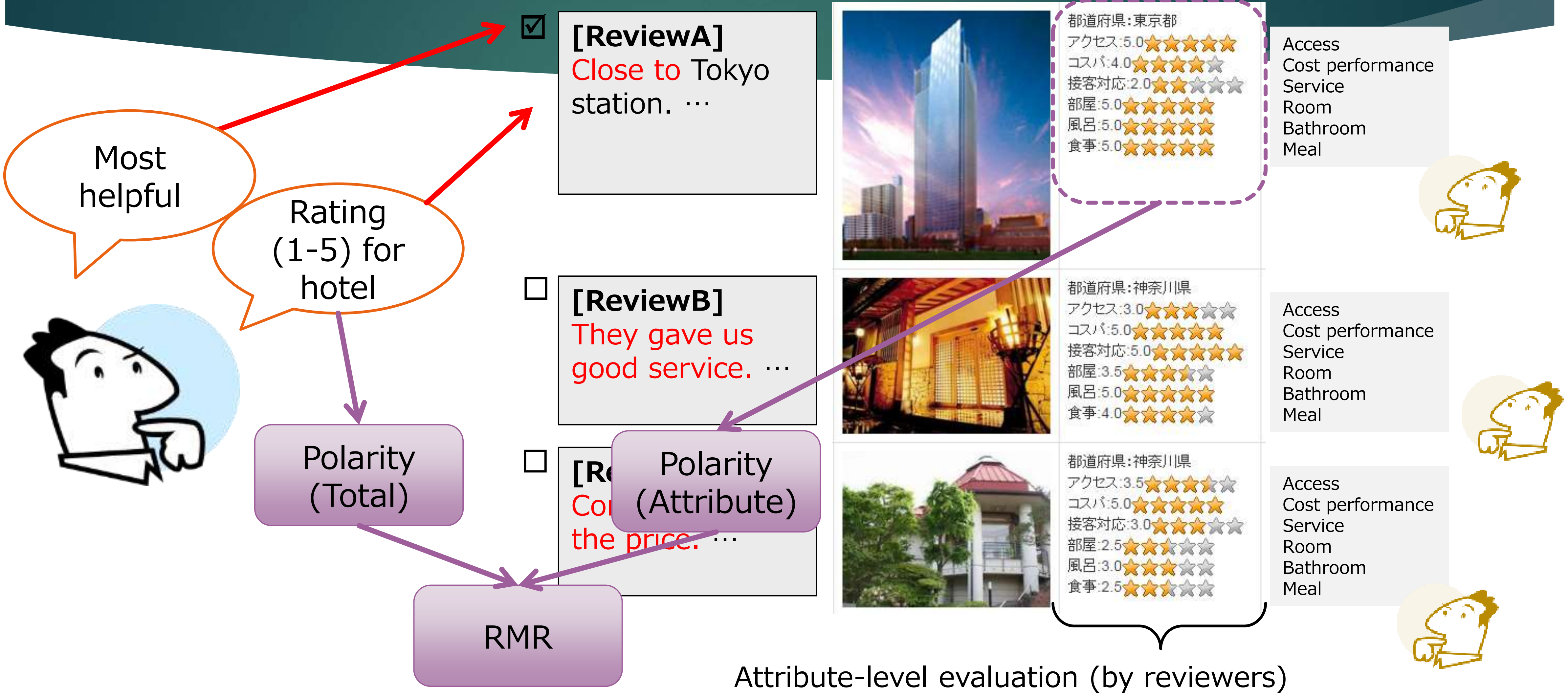


MCFPV(PRE) MCFPV(BPR) SVD PMF NMF

Good / Bad Points of Personal Values-based User Modeling

- [GOOD] Model is constructed on attribute space of target item
  - Easy to combine with ordinary recommendation methods
  - Can be calculated for any attribute IF rating is given
- [GOOD] Stable modeling with small number of reviews (<10)
  - Effective for cold-start / sparsity problem
- [BAD] Need reviews POSTED by target users
  - # of reviewers << # of ROMs

User Modeling from Review Browsing Behavior



From user modeling to item modeling

User modeling

Rating records of target user

Total	★★★★★
Story	★★★★★

Total	★★☆☆☆
Story	★★★★☆

⋮

RMR

[Proposed] Item modeling

Rating records of target item

Total	★★★★☆
Story	★★★★★

Total	★★★☆☆
Story	★★☆☆☆

⋮

Lift value

More review available for item than user

From RMR to Lift value

Personal-values-based user model

Attribute evaluation	Total evaluation
Pos	Pos
Neg	Neg

$$\text{RMR} = \frac{\#matched}{\#unmatched + \#matched}$$

Proposed method

Attribute evaluation	Total evaluation
Pos	Pos
Pos	Neg
Neg	Pos
Neg	Neg

Lift value

Calculate 4 values for attribute

Calculation of Lift value

X: Polarity of Attribute evaluation

Y: Polarity of Total evaluation

X		Y
Pos	➡	Pos
Pos	➡	Neg
Neg	➡	Pos
Neg	➡	Neg

4 patters of lift value

$$\mathrm{lift}(X \rightarrow Y) = \frac{\mathrm{P}(X \wedge Y)}{\mathrm{P}(X)\mathrm{P}(Y)}$$

Example for movie data

Attr	P→P	P→N	N→P	N→N
Story	2.00	0.67	0.00	1.33

$$\mathrm{lift}(Pos \rightarrow Pos) = 2.0$$

The probability of "*The movie is favored*" doubles with the condition of "*Story is favored*"

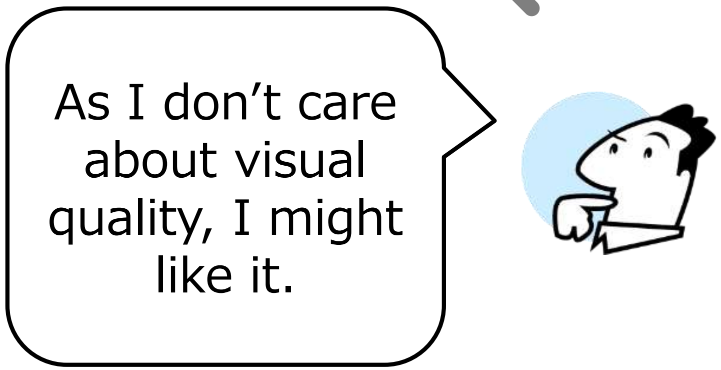
Explaining recommendation with lift value

Attribute evaluation	Total evaluation
Pos	Pos
Pos	Neg
Neg	Pos
Neg	Neg

Attribute	P→P	P→N	N→P	N→N
Story	2.00	0.67	0.00	1.33
Casts	1.08	0.93	0.87	1.11
Direction	1.22	0.81	0.83	1.14
Visual quality	0.00	1.33	2.00	1.33
Music	1.12	0.67	0.97	1.09

"People who like story tend to be satisfied with the movie"

"People tend to be satisfied with the movie even though they do not like Visual quality"



Conclusion

- Personal values-based information recommendation
  - RMR: Modeling user's personal values
  - Introduction to collaborative filtering (neighborhood-based, Matrix-based): effective for long-tail item recommendation
  - User modeling from browsing history
  - Item modeling with explanation
- Beyond recommending favorite items
  - Paradigm shift to acceptable items
  - Extend applicability of recommender systems: behavior change support, etc.