大阪大学
OSAKA UNIVERSITY

UNIVERSITY OF MINNESOTA
Driven to Discover℠

Offline Evaluation for Recommender Systems

Osaka University
Graduate School of Engineering Science
Yoshinori Hijikata

This work has been done while the author worked for GroupLens Research in University of Minnesota as visiting scholar in 2014.

Table of contents

How to evaluate RS

- Evaluation of recommender systems (RS)
  - Various kinds of accuracy (correctness) metrics
    - Evaluate system's predicting ability or accuracy of recommendation list?
    - Correct data is given by n-point scale or binary (unary) scale?
  - Incorporating the novelty and serendipity
    - Whether unknown items are recommended?
    - Recommendation gives users surprise?

Need to summarize the existing evaluation metrics

Objective of this slide

- Cover almost all the existing offline evaluation metrics of RS (Not only accuracy metrics but also discovery metrics)
- Categorize them from the evaluation goal
- Make the explanation easy and simple (Do not insist readers to consult other textbooks or papers)

Overview of Offline Evaluation for Recommender Systems

Overview of offline evaluation

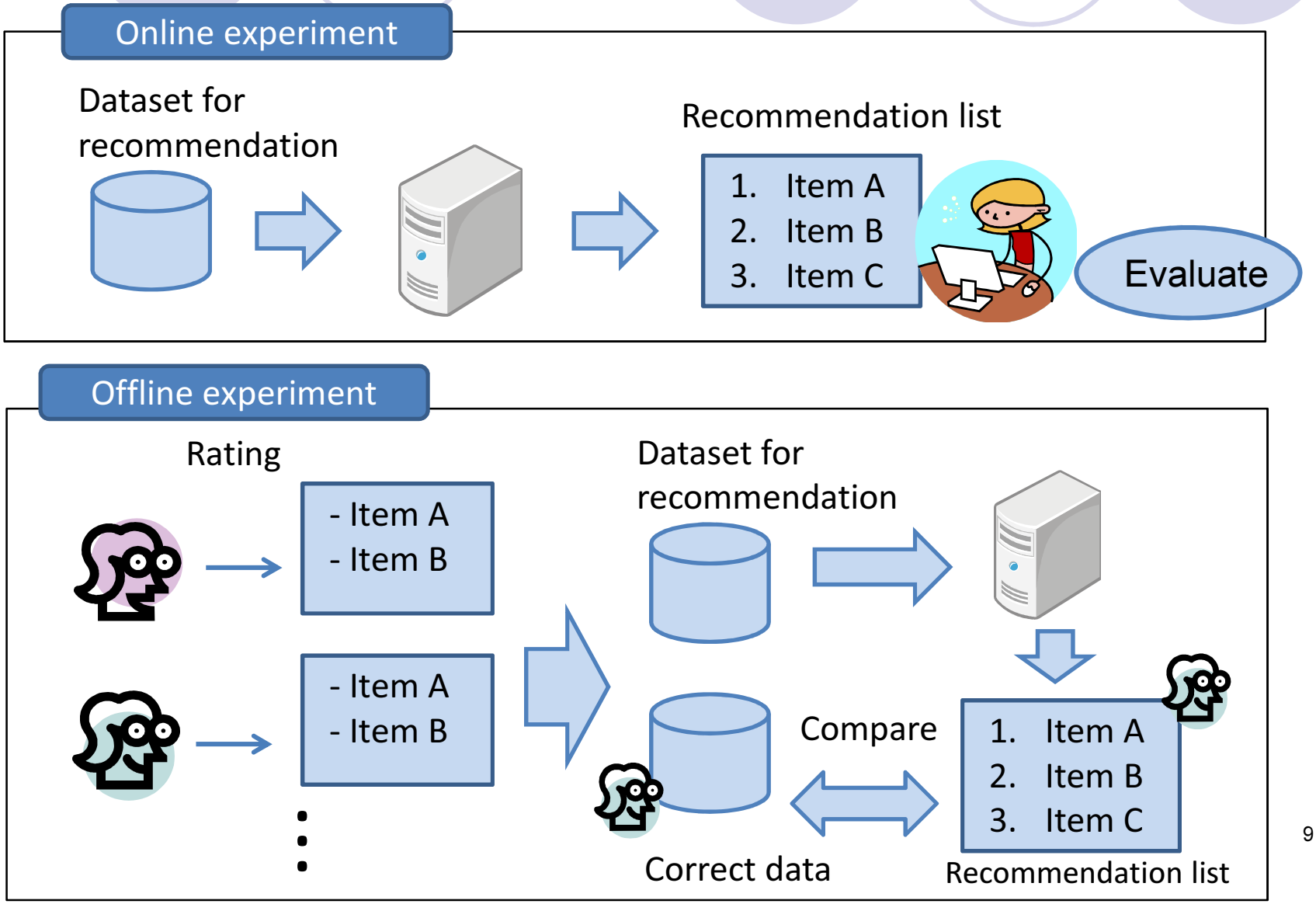
- Online vs. offline
- Characteristics of dataset to be considered
- Dataset and its separation for offline experiment

Online vs. Offline evaluation

Outline of online and offline

- Online evaluation [Herlocker 04, Gunawardana 09]
  - Also called “online experiment” or “live user experiment”
  - Let users use the RS and examine their performance to the tasks
- Offline evaluation
  - Also called “offline experiment” or “offline analysis”
  - Collect users’ ratings to items in advance. Some of the ratings are for training the RS and others are for evaluating it.

Outline of online and offline



Overview: Online vs. Offline

	Overall evaluation	Reproducibility	Measurement consistency	Preparation cost
Online	Good	Bad	Bad	Bad
Offline	Bad	Good	Good	Good

	Extensibility	Time sensitivity	Further analysis	Stabililty	Scalability
Online	Good	Good	Good	Bad	Bad
Offline	Bad	Bad	Bad	Good	Good

Characteristics of evaluation

- Overall evaluation
  - Whether the method can directly evaluate the entire system.
    - Online evaluation can evaluate it by directly asking to users.
    - Offline evaluation cannot evaluate it because it has only the information about the user's evaluation to items.
- Reproducibility
  - Whether other researchers can reproduce the same setting of experiment.
    - It is easy in offline evaluation because they usually use the same dataset (open dataset)
    - It is difficult in online evaluation because they give more complex instructions to users and measure the real-time behaviors.

Characteristics of evaluation

- Measurement consistency
  - Whether the meaning of metrics are commonly recognized among researchers.
    - It is highly consistent in offline evaluation because they use the same dataset and the user's task is simple.
    - It is low consistent in online evaluation because there is a variety in users' tasks.
- Preparation cost
  - How long the experimenter takes time, how much they take efforts for the preparation.
    - There is no preparation cost in offline evaluation when they use the open dataset.
    - Preparation cost is high in online evaluation because they set details of users' tasks, questionnaires, metrics.

Characteristics of evaluation

- Extensibility
  - Whether they can add new evaluation metrics.
    - It is difficult to add metrics (beyond-accuracy metrics) in offline evaluation because the dataset is usually fixed.
    - It is easy to add new metrics in online evaluation because they can ask any questions to users.
- Time-sensitivity
  - Whether they can evaluate the system's performance with time (at any time)
    - It is difficult to analyze with time passed in offline evaluation because they cannot run the system in real-time.
    - It is easy to analyze with time passed in online evaluation because they can measure users' real-time behaviors.

Characteristics of evaluation

- Further analysis
  - Whether experimenters can analyze deeply the results considering the users' internal status*.
    - It is difficult to analyze deeply in offline evaluation because they cannot ask questions about users' internal status.
    - It is easy to analyze deeply in online evaluation because they ask any questions about users' internal status.

* If the experimenter asked users about their internal status, they can apply deep analysis method like path analysis or structural equation modeling. [Bollen 10, Ekstrand 14]

Characteristics of evaluation

- Stability
  - Whether the user's evaluation is stable among different timing to ask.
    - It is relatively stable* in offline evaluation because the questionnaire is simple.
    - It is not stable in online evaluation because users answer questionnaires after using the system (in different contexts).

* Even in offline evaluation, users' ratings to items are not stable. Users may give different rating value to the same item if the timing of the questionnaire is different. [Hill 95, Cosley 03, Amatriain 09]

Characteristics of evaluation

- Scalability
  - Whether the experimenter conducts an experiment with many users and items.
    - It is relatively easy to collect many users and items in offline evaluation because the task is simple.
    - It is difficult to collect many users and items in online evaluation because the task is complex (Users have to use the system in different contexts).

Characteristics of dataset to be considered

Characteristics of dataset

- Explicit or implicit
  - It is more reliable if the system directly elicits the user's interest or preference from the user.
- Scaling
  - The degree of scale granularity. (eg. unary, binary, 3-point scale, 5-point scale, or more) (Usually, Likert scale)
- Rating bias
  - Users' tendency to rate items toward highly or low.
    - Generally, higher bias in ratings. [Kamishima 07]
- Timestamp
  - Whether each rating has timestamp?

Characteristics of dataset

- Multi-criteria ratings
  - Single rating criterion or several criteria to item.
    - e.g. food, decor, service for restaurant review

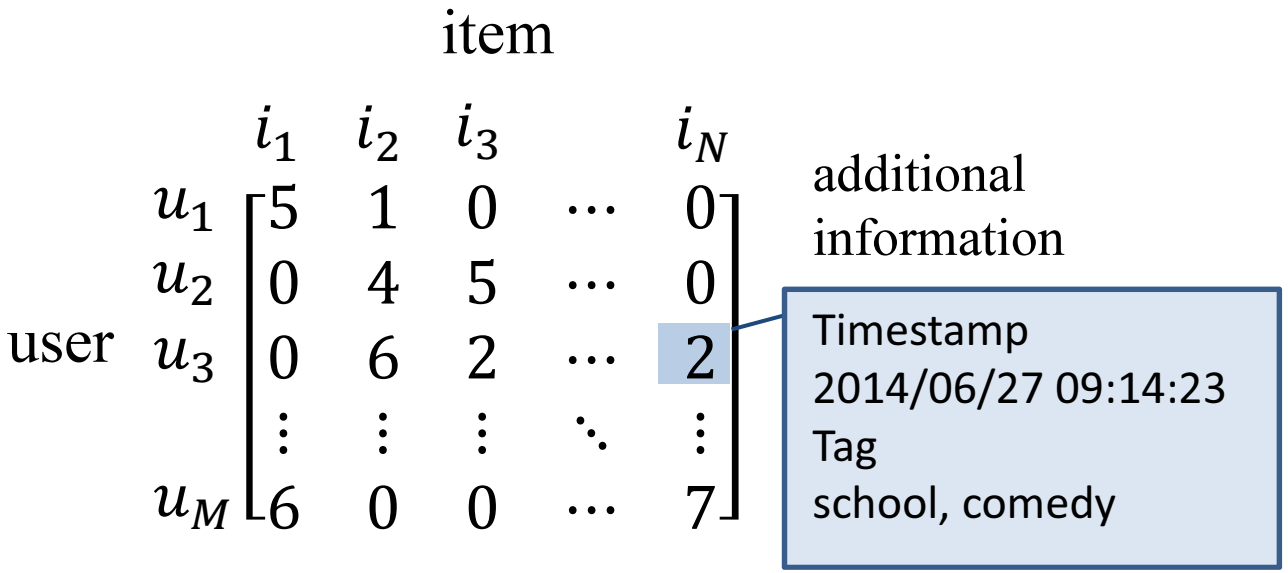
[Admavicius 07]

- Data size
  - How large about the number of users and items?
- Density
  - How sparse the rating matrix is?
- Data increment
  - How frequently the new data is input?

Dataset and its separation for offline evaluation

Dataset for offline evaluation

- Consist of user's rating value to item
  - e.g. 7-point scale. 0: no rating
  - Some dataset has timestamp or tag



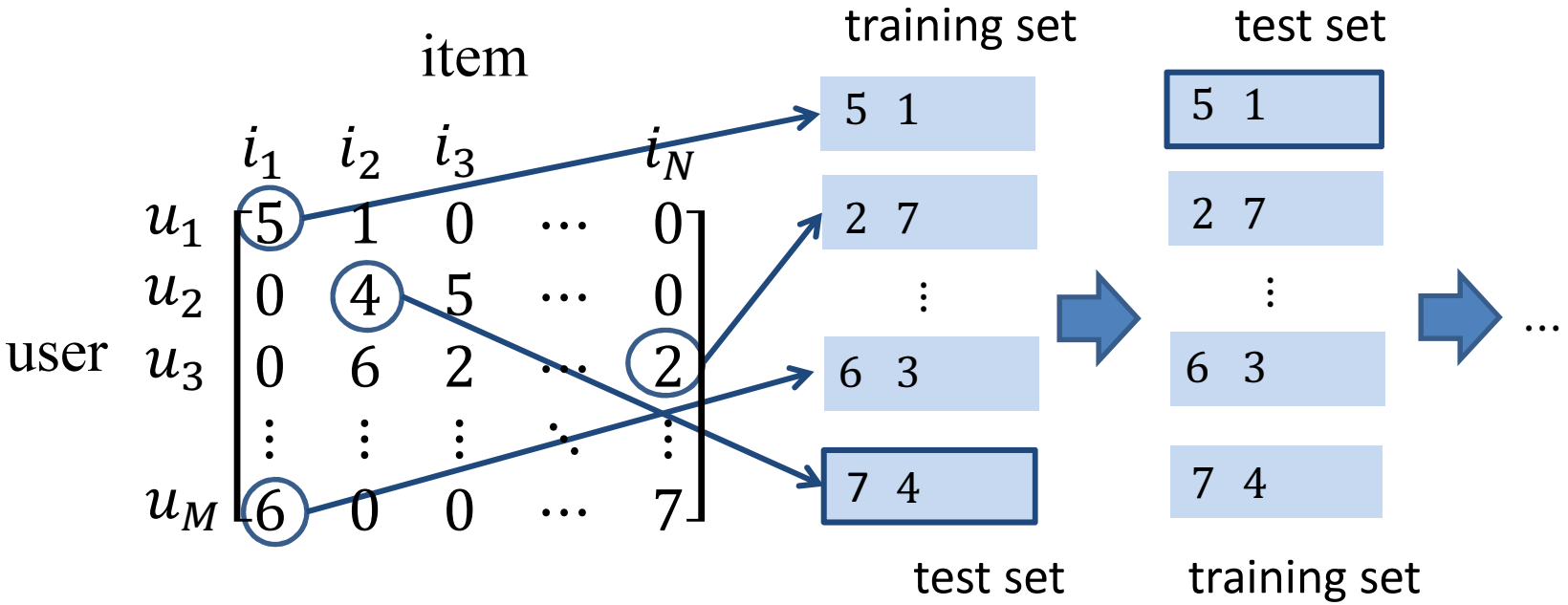
Dataset for offline evaluation

- Feature data for content-based filtering
  - RS with content-based filtering needs the feature data of items [Adomavidius 05, Lops 11]

$$
\begin{array}{cc} & \text{feature} \\ & \begin{array}{ccccccc} f_1 & f_2 & f_3 & f_4 & f_5 & & f_L \end{array} \\ \text{item} & \begin{array}{c} i_1 \\ i_2 \\ \\ i_N \end{array} \begin{bmatrix} 0.2 & 0.7 & 0.1 & 0.0 & 0.2 & \cdots & 0.2 \\ 0.4 & 0.1 & 0.6 & 0.5 & 0.3 & \cdots & 0.9 \\ & & & \vdots & & & \\ 0.0 & 0.5 & 0.3 & 0.5 & 1.0 & \cdots & 0.3 \end{bmatrix} \end{array}
$$

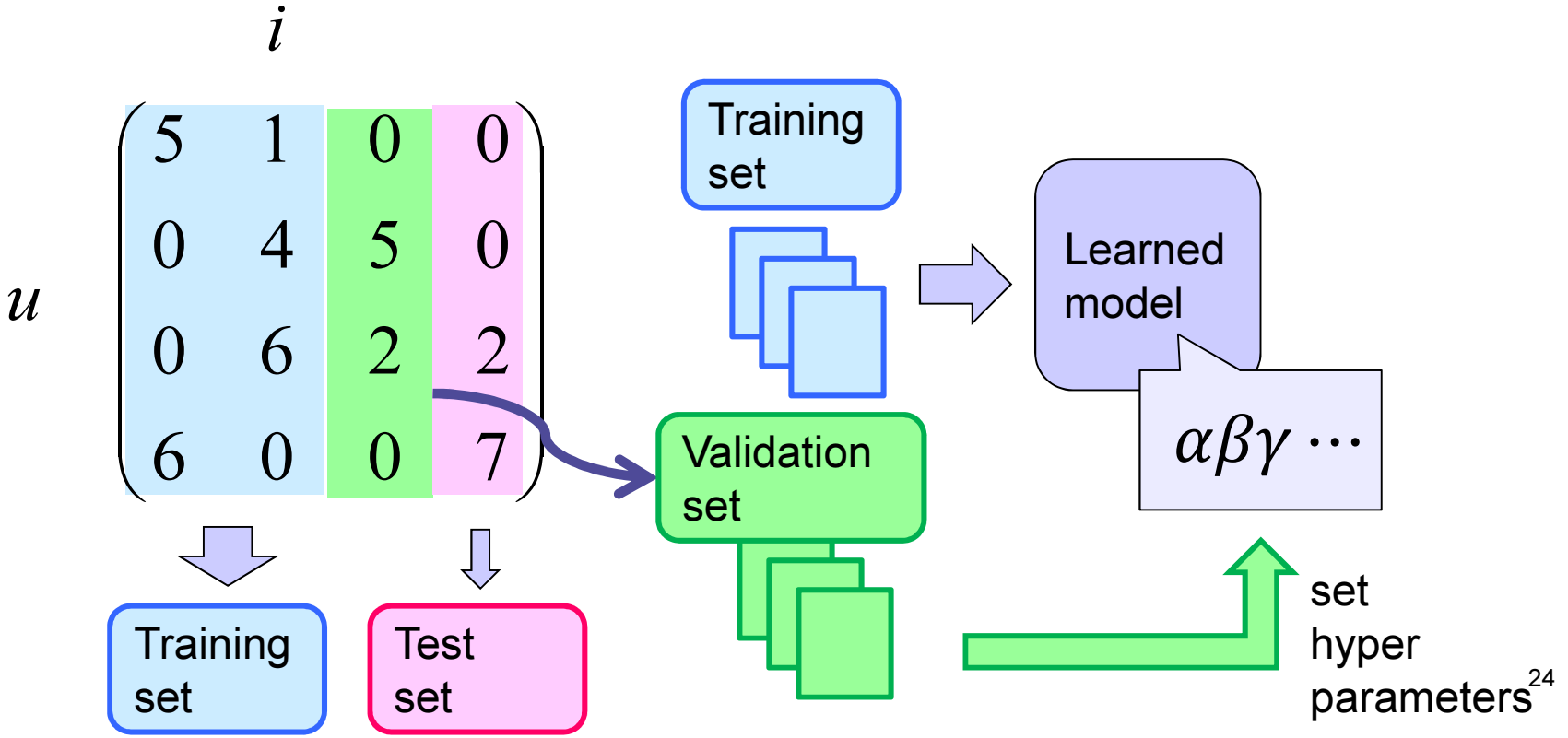
Cross validation

- Cross validation (k-fold cross validation) [Stone 74]
  - Dataset are separated to K groups.
  - One is for test set and the others are for training set
  - Replace the group for test set



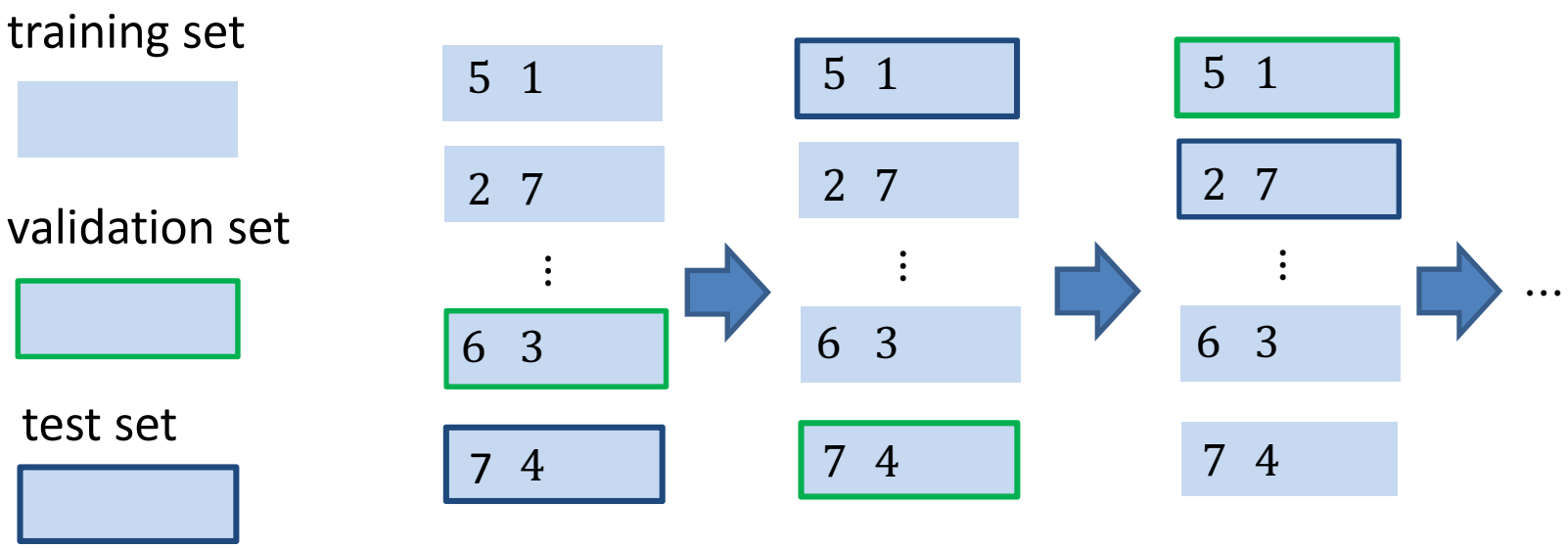
Cross validation

- Validation (development, tuning) set
  - Some algorithms need to be set hyper parameters
  - Prepare one data group for testing the parameters



Cross validation

- Cross validation with validation set

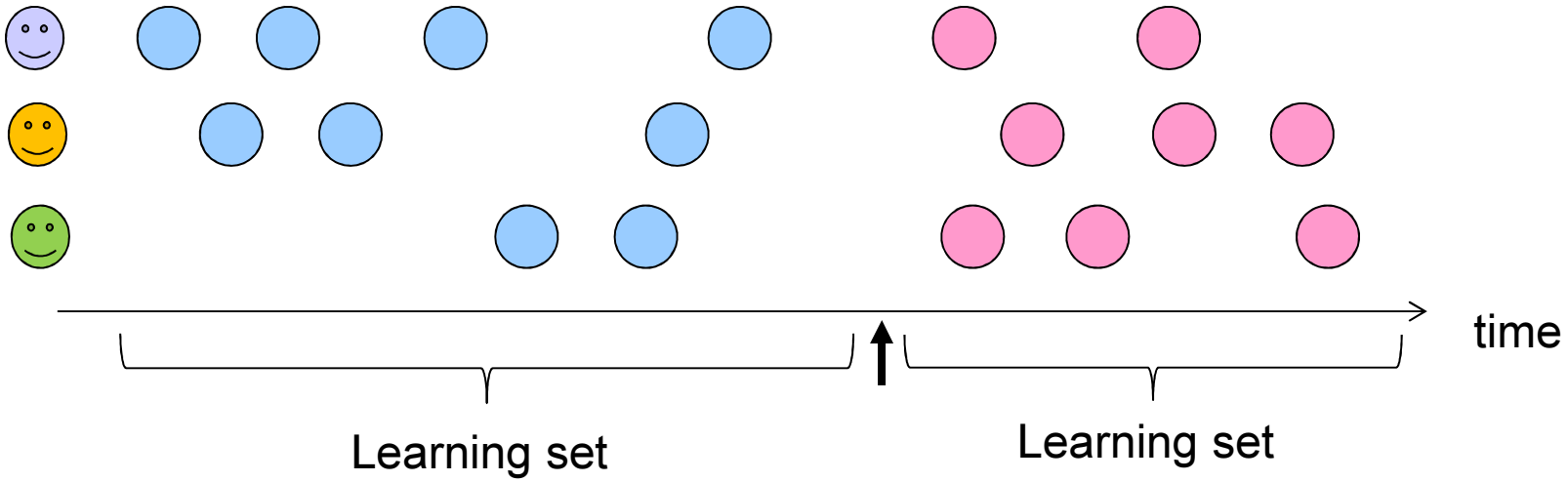


Data separation for timestamp data

- Data separation for time stamp data

[Gunawardana 09]

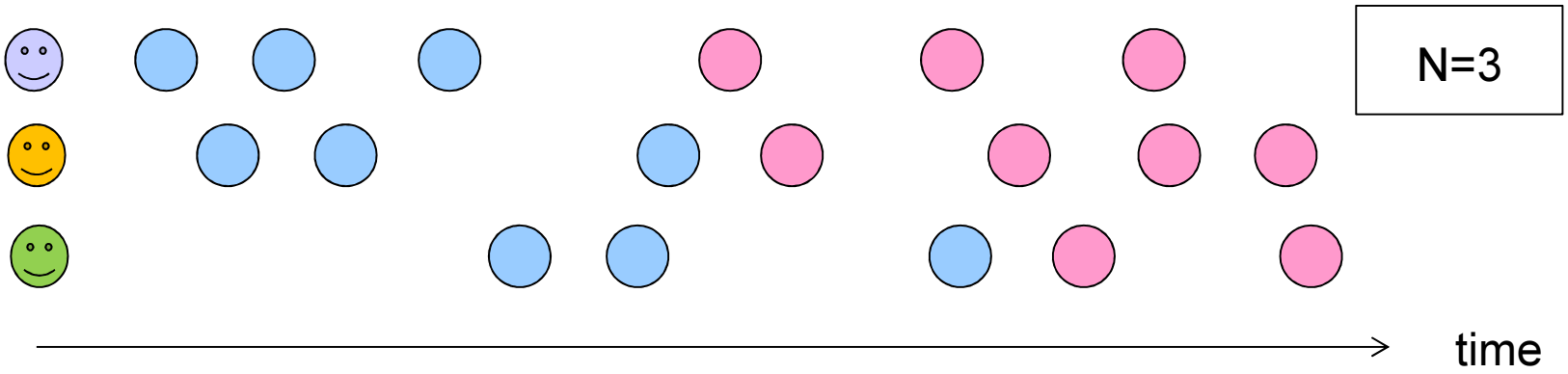
(1) Fixed separation time for all users



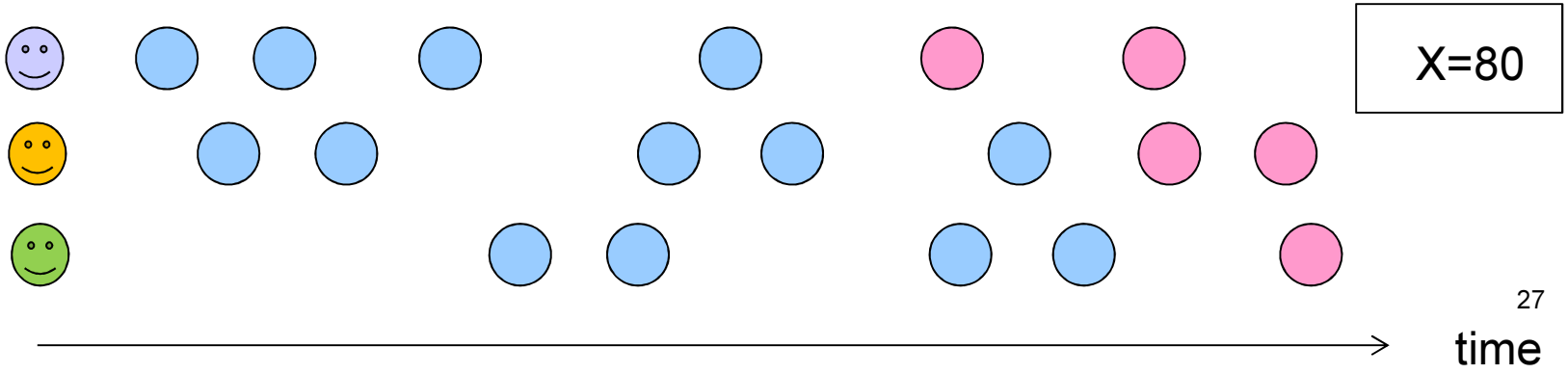
Data separation for timestamp data

(2) Separated for each user

(2-1) The former N data for learning set, the latter data for test set

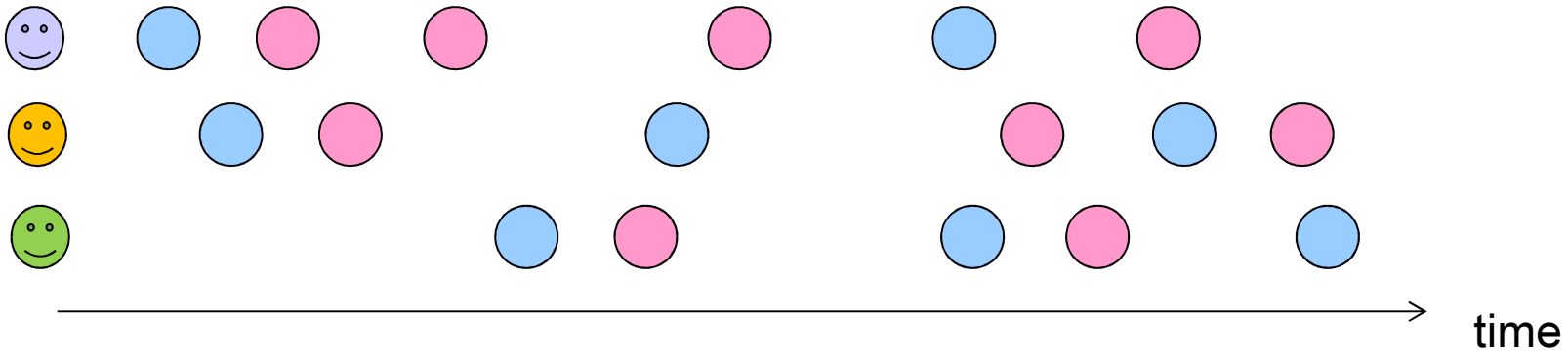


(2-2) The former X% data for learning set, the latter data for test set

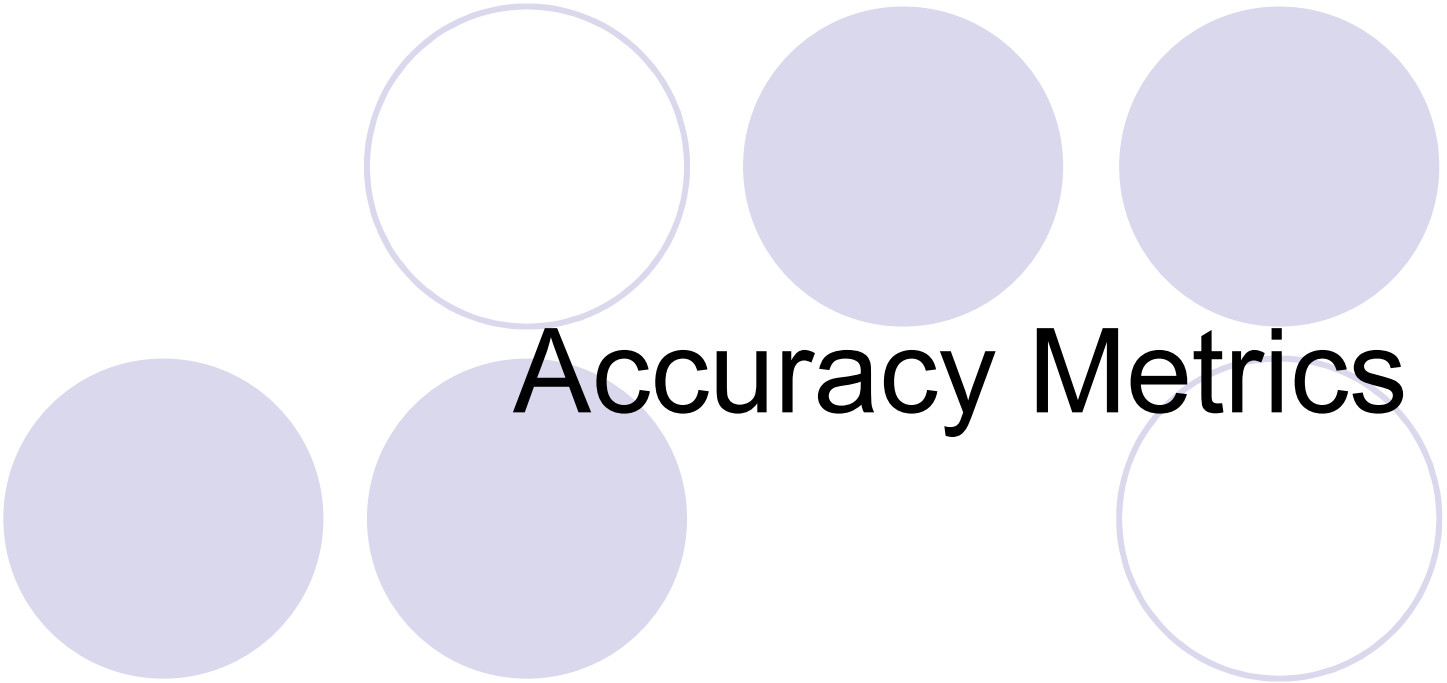


Data separation for timestamp data

(3) Random separation



*In random separation, k-fold cross validation can be applied to.



Accuracy Metrics

Introduction: Accuracy metrics

- Accuracy metrics
  - Consider whether the item is fit for the user's interest or preference
  - Do not care whether the recommended item is useful for the user
  - Good recommendation should have high accuracy [Sinha 01]
  - The user satisfaction is strongly influenced by the accuracy [Hijikata 12, Ekstrand14]

*My definition is different from the accuracy (Rand accuracy, Rand index) in machine learning area.

Types of accuracy metrics

- Accuracy metrics
  - Accuracy of estimated rating
  - Accuracy of estimated ranking
  - Accuracy of list relevance
  - Accuracy based on ranking position

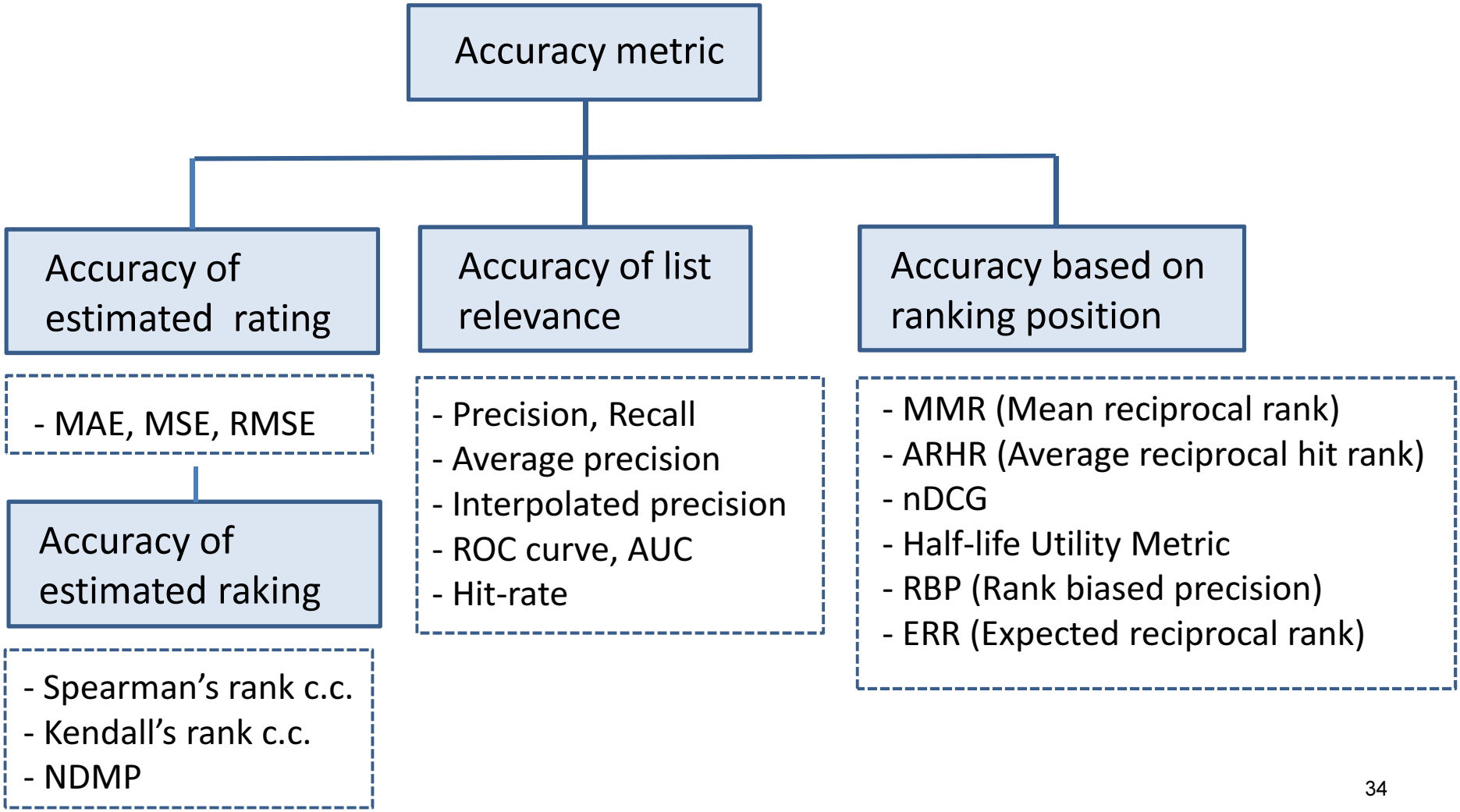
Overviews of accuracy metrics

- Accuracy of estimated rating
  - RS estimates the user's rating value.
  - Measure the difference between estimated value and correct value (given by user)
- Accuracy of estimated ranking
  - RS orders the item to be shown to user.
  - Measure the correctness of the order by comparing with the correct order (given by user)

Overviews of accuracy metrics

- Accuracy of list relevance
  - RS produces a list of items as recommendation.
  - Measure the relevance of each items to the user's preference
- Accuracy based on ranking position
  - Highly relevant item should be at high rank, lower one should be at low rank.
  - Measure the list relevance considering its ranking position

Accuracy metrics and categorization



Accuracy of estimated rating

MAE, MSE, RMSE

- MAE (Mean absolute error)

$$MAE = \frac{\sum_{b \in B} |r(b) - p(b)|}{|B|}$$

B : Item set
p : prediction
r : correct rating

Ground truth

$$\begin{pmatrix} 5 & 1 & 0 & 6 \\ 0 & 4 & 4 & 0 \\ 0 & 6 & 2 & 2 \\ 6 & 0 & 0 & 7 \end{pmatrix}$$

learning data | test data

System prediction

$$\begin{pmatrix} - & - & - & 3 \\ - & - & 6 & - \\ - & - & 1 & 5 \\ - & - & - & 6 \end{pmatrix}$$

how much different?

MAE, MSE, RMSE

(For evaluating all test set data)

- MSE (Mean square error)

$$MSE = \frac{\sum_{b \in B} |r\,(b) - p\,(b)|^2}{|B|}$$

Consider the large difference more serious

B : Item set

p : prediction

r : correct rating

- RMSE (Root mean square error)

$$RMSE = \sqrt{= \frac{\sum_{b \in B} |r\,(b) - p\,(b)|^2}{|B|}}$$

Make it as same unit as MAE

MAE, MSE, RMSE

- Normalization

$$normalizedMAE = \frac{MAE}{r_{max} - r_{min}}$$

r_{max} : Maximized value

r_{min} : Minimized value

$$normalizedMSE = \frac{MSE}{r_{max} - r_{min}} \qquad normalizedRMSE = \frac{RMSE}{r_{max} - r_{min}}$$

- Pros and Cons
  - [Pros] Can evaluate all the items in the test set
  - [Cons] Cannot distinguish differences in lower rating and medium rating.

    (e.g. rating 1-2 and rating 2-3 (neutral))
  - [Cons] Users cannot perceived the small difference [Lam 06].

Accuracy of estimated ranking

Accuracy of estimated ranking

Ground truth

item i

		1	2	3	4	5	6	7	8	9	10	11	12	13
user u	[	3	4	2	0	4	7	6	3	1	5	2	4	2]

System prediction

item i

		1	2	3	4	5	6	7	8	9	10	11	12	13
user u	[	3	4	2	0	2.6	5.3	5.6	4.5	5.5	6.2	3.5	4.1	1.5]

Ranking from the head to the tail

Evaluate this ranking

System ranking

1. item 10
2. item 7
3. item 9
4. item 6
5. item 8
6. item 12
7. item 11
8. item 5
9. item 13

User ranking

1. item 9
2. item 12
3. item 13
4. item 8
5. item 5
6. item 12
7. item 10
8. item 7
9. item 6

Spearman's ranking correlation

- Spearman's rank correlation
  - Non parametric
  - Suited for measuring ranking validity
- Pearson correlation
  - Parametric: Suppose bivariate normal distribution between variables

user's rating

[1.0 2.0 3.0 4.0 5.0 6.0 7.0 8.0 9.0 10.0]

system rating

[1.0 2.0 3.0 4.0 5.0 6.0 7.0 8.0 9.0 10.0]

Peason r

$r = 1.0$

[1.0 1.5 2.0 2.5 3.0 8.0 8.5 9.0 9.5 10.0]

Rankings are the same.

Spearman's ranking correlation

- Calculation method of Spearman r

Pearson r

$$r_{Pearson} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

↓ Use rank instead of using observed value

$$\sum x_i = n(n+1)/2 \qquad \sum x^2 = n(n+1)(2n+1)/6 \qquad \bar{x} = (n+1)/2$$

Spearman r

$$r = 1 - \frac{6}{n(n^2-1)}\sum_{i=1}^{n}(x_i - y_i)^2$$

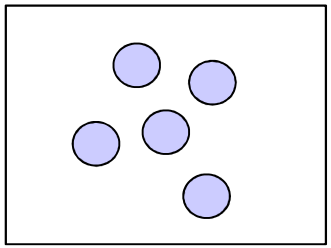
x : Ranks of item i output by RS

y : Ranks of item i offered by user

Kendall's rank correlation coefficient

- Kendall's rank correlation τ
  - Check two users' coincidence about magnitude relationship between two items

Item set



for all item pairs (a, b)

$n(n-1)/2$

User s and t give ranking to items

IF $Rank_s(a) < Rank_s(b)$ AND $Rank_t(a) < Rank_t(b)$ THEN $P \leftarrow P + 1$

IF $Rank_s(a) > Rank_s(b)$ AND $Rank_t(a) > Rank_t(b)$ THEN $P \leftarrow P + 1$

OHTERWISE $Q \leftarrow Q + 1$

$$\tau = \frac{P - Q}{\frac{1}{2}n(n-1)} = \frac{2P}{\frac{1}{2}n(n-1)} - 1$$

NDPM

- NDPM (Normalized distance-based performance measure)

$$NDPM = \frac{2C^{-} + C^{u}}{2C^{i}}$$

C^{-} : The number of contradictory preference relations which happen when the system says item 1 will be preferred to item 2, but the user ranking is opposite

C^{u} : The number of compatible preference relations which happen when item 1 will be preferred to item 2 in the system's ranking, the user sees them equal

C^{i} : The total number of preferred relations (the same order) of item pairs between the system's and the user's ranking

System's ranking

Rank	Item
1	Item A
2	Item B
3	Item C
4	Item D
5	Item E

User's ranking

Rank	Item
1	Item A
2	Item B
2	Item C
4	Item E
5	Item D

C^{u} C^{-}

[Yao 95]

Spearman, Kendall, NDMP

- Pros and Cons
  - [Pros] Evaluate the rankings for both the recommendation list and the whole test set
  - [Cons] Difficult to obtain the complete ranking of the active user
  - [Cons] NDPM reduces the penalty if the system gives different ranks to items with the same ranks in user evaluation (This case will always happen in RS)

Accuracy of list relevance

Accuracy of list relevance

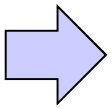
- Probability Rank Principle (PRP) [Robertson 97]
  - The relevance of a document to a query is independent of the relevance of other documents the user has seen before.
  - The utility will be maximized when the system orders documents according to their relevance to the query.
  - Most IR and RS follow this principle, and the result will be presented to the user in a list.

Accuracy of list relevance

- Precision, recall and F-value

user's rating

	1	2	3	4	5	6	7	8	9	10
u	5	1	0	3	6	7	1	2	1	7

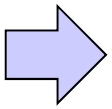


user's rating

	1	2	3	4	5	6	7	8	9	10
u	5	1	-	U	F	F	U	U	U	F

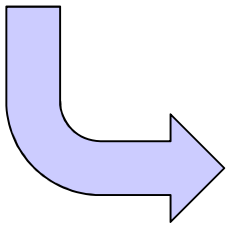
system's prediction

	1	2	3	4	5	6	7	8	9	10
u	5	1	0	1	7	5	1	6	5	3



system's prediction

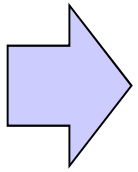
	1	2	3	4	5	6	7	8	9	10
u	5	1	U	U	F	F	U	F	F	U



Rank	Item
1	Item 5
2	Item 8
3	Item 6
4	Item 9

User's rating

F
U
F
U



Precision 2/4=50%
Recall 2/3=67%

Precision, Recall, F-measure

- Precision

$$Precision_i = \frac{|T_i \cap \supset L_i|}{|\supset L_i|}$$

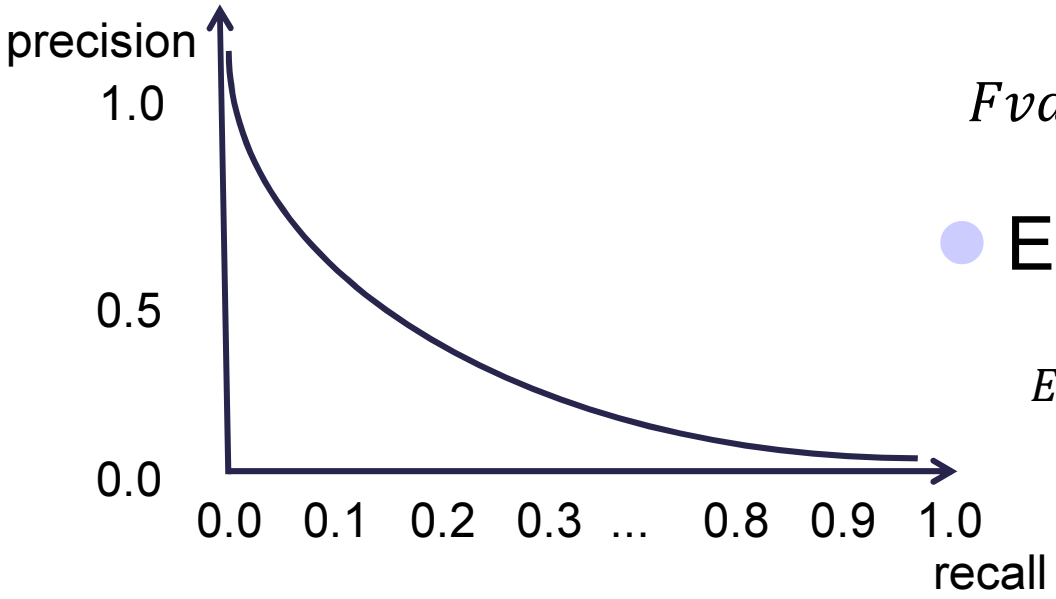
- Recall

$$Precision_i = \frac{|T_i \cap \supset L_i|}{|\supset L_i|}$$

T_i : User i's favorite item set in the test set

L_i : System's recommendation list

- Precision recall curve



- F-measure

$$Fvalue_i = \frac{2 \cdot Precision_i \cdot Recall_i}{Precision_i + Recall_i}$$

- E-measure

$$Evalue_i = 1 - \frac{1 + b^2}{\frac{b^2}{Recall_i} + \frac{1}{Precision_i}}$$

Average precision, MAP

- Average precision

$$AP = \frac{1}{M} \sum_{1 \le k \le N} rel(k) \cdot Prec@k$$

M : The number of matched items in the list

N : The list length

$rel(\,)$: A function returning matched or not

- MAP (Mean average precision)

$$MAP = \frac{1}{m} \sum_{m} AP_m$$

m: The number of recommendation list (The number of test queries in IR)

[Buckley 05]

Interpolated Precision

- Interpolated Precision [Manning 08]

Rank	1	2	3	4	5	6	7	8	9	10
Fit?	Y	N	Y	N	Y	Y	N	N	N	N
Precision	1.0	0.5	0.67	0.5	0.6	0.67	0.57	0.50	0.44	0.40
Recall	0.25	0.25	0.5	0.5	0.75	1.0	1.0	1.0	1.0	1.0

1. Find the highest precision $precision_h$.
2. Find the recall $recall_h$ at that time.

Recall	- 0.25	0.25 - 1.0
Interpolated Precision	1.0	0.67

3. Consider $precision_h$ as the precision at the recall smaller than $recall_h$
4. Repeat 1.

n-points Interpolated Precision

- n-points interpolated average precision

  Set N-points recall.

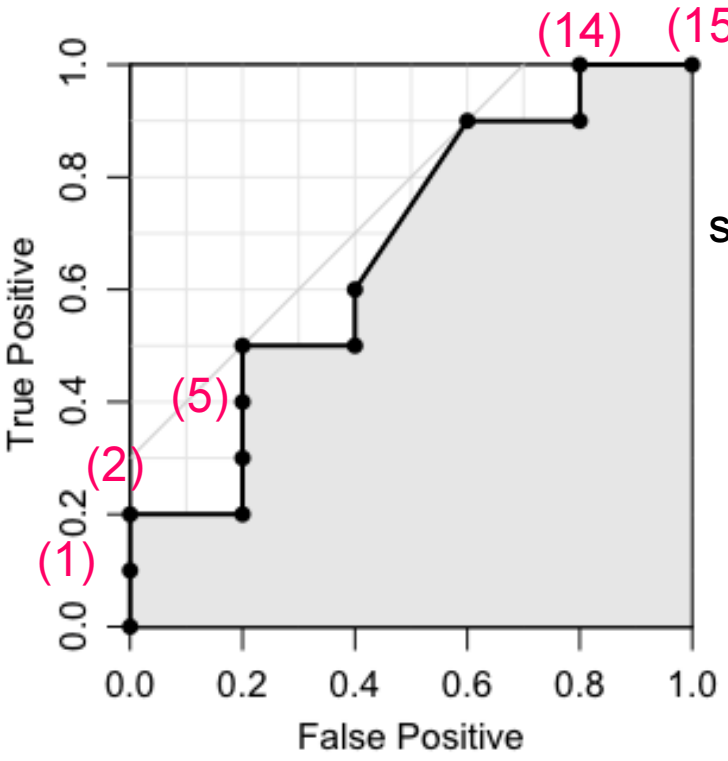
  Usually N=11 (recall 0.0 0.1 0.2 ... 0.9 1.0)

$$nAIP = \frac{1}{N}\sum_{i} IntPrecision_i$$

ROC curve

- ROC curve (receiver operating characteristic)

Like: The user prefers the item in the ground truth data
Dislike: The user dislikes the item in the ground truth data
positive: The system determines the item as favorite one
negative: The system determines the item as un-favorite one



user

system	Like	Dislike
positive	TP	FP
negative	FN	TN

$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$

$$RandAccuracy\ (Index) = \frac{TP + TN}{TP + FP + TN + FN}$$

	P-value	F/N
(1)	0.95	L
(2)	0.85	L
	0.80	D
	0.75	L
(5)	0.70	L
	0.65	L
	0.55	D
	0.45	L
	0.30	L
	0.30	L
	0.30	L
	0.30	D
	0.18	D
(14)	0.15	L
(15)	0.05	D

GROC curve and CROC curve

- How to obtain ROC curve from several users' logs
  - Global ROC curve (GROC curve) [Schein 02]
    - Calculate the prediction score to all the pairs of user and item in the test set
    - Order the pairs in descending order and make a list
    - Draw ROC curve
  - Customer ROC curve (CROC curve) [Sarwar 00]
    - Create recommendation list in each user
    - Calculate TPR and FPR in each user
    - Take the above average and draw ROC curve

Hit-rate

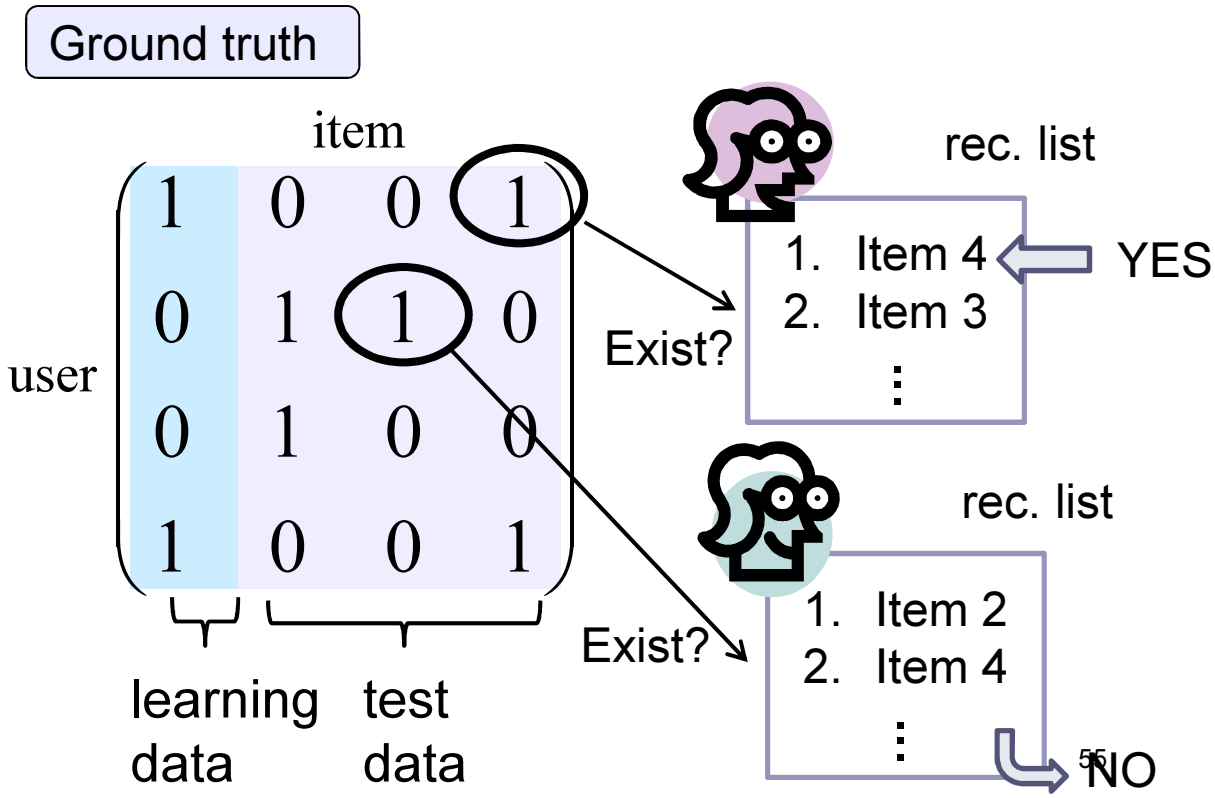
- Hit-rate [Deshpande 04]
  - Target unary data (purchase logs, viewing logs)

1. Search non-zero entry
2. Check whether the item included in the list
3. Calculate hit-rate

$$hit-rate = \frac{num_{hits}}{n}$$

num_{hits} : items included in the rec. list

n : The number of users



Accuracy based on ranking position

Accuracy based on ranking position

- Accuracy based on ranking position
  - Evaluate ranking list considering is relevant item's ranking position [Craswell 08, Richardson 07]
  - Position-based model

$$P(C = 1|i, l) = atr_i \cdot p_l \quad [\text{ Chapelle 09 }]$$

i : item l : Ranking position C : Event that the user watched the item

atr_i : Item i's attractiveness to the user

p_l : Provability that the user checked until ranking

    - MRR, APHR, DCG, nDCG, Half-life Utility metric, RBP
  - Cascade-based model [Chapelle 09]
    - Consider the user's stop browsing with their satisfaction
    - ERR

Mean Reciprocal Rank (MRR)

- Mean Reciprocal Rank (MRR)
  - Take average of reciprocal of ranks

$$MRR = \frac{1}{|D_{rel}|}\sum_{i=1}^{N}\frac{rel_i}{i}$$

D_{rel} : The set of items preferred by the user

N : The length of the recommendation list

rel_i : Whether the item is preferred by the user (1 or 0)

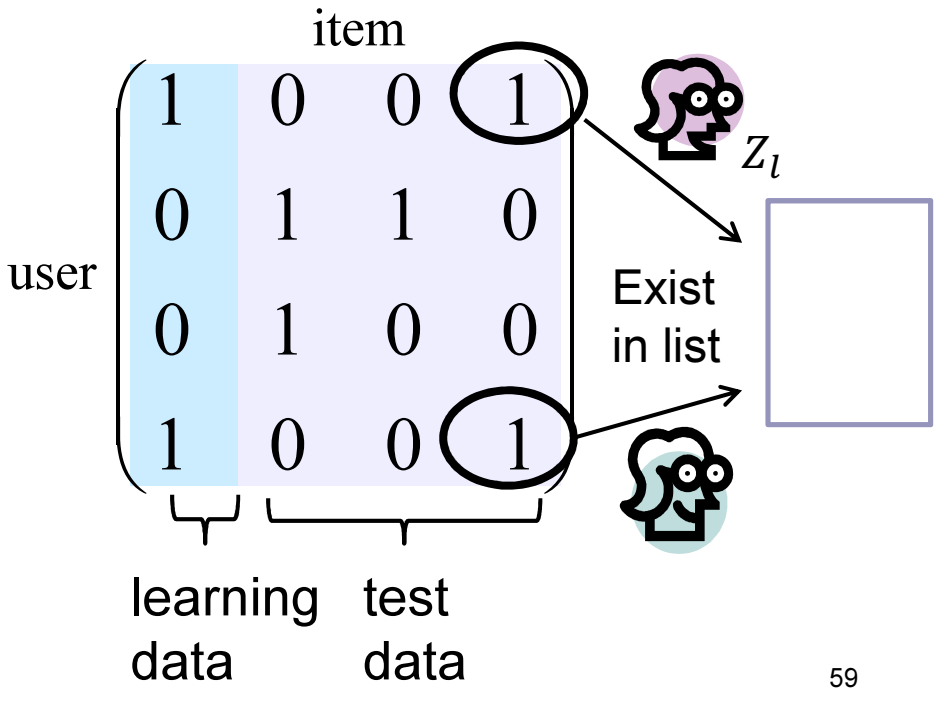
ARHR

- ARHR (Average Reciprocal Hit-Rank) [Deshpande 04]
  - Target unary data (purchase logs, viewing logs)

$$ARHR = \frac{1}{n}\sum_{i \in Z_l} \frac{1}{pos(i)}$$

Z_l : Non-zero entry included in the user's rec. list

$pos()$: Item i's ranking position



CG, DCG, nDCG

- Cumulative gain (CG)

$$CG_p = \sum_{i=1}^{p} rel_i$$

p : List length

rel_i : User's actual rating

- Discounted cumulative gain (DCG)

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{log_2(i)}$$

$$DCG_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{log_2(i+1)}$$

Rank	Rel
1	0.95
2	0.23
3	0.85
4	0.90
5	0.88
6	0.67
7	0.56
8	0.45
9	0.35
10	0.92

[Jarvelin 02]

CG, DCG, nDCG

- Normalized DCG (nDCG)

$$IDCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{log_2(i)}$$

for ideal order

$$IDCG_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{log_2(i+1)}$$

$$nDCG_p = \frac{DCG_p}{IDCG_p}$$

For considering variety of recommendation length

Ideal order

Rank	Rel
1	0.95
2	0.92
3	0.90
4	0.88
5	0.85
6	0.67
7	0.56
8	0.45
9	0.35
10	0.23

[Jarvelin 02]

Half-life utility metric

- Half-life utility metric

[Breese 98]

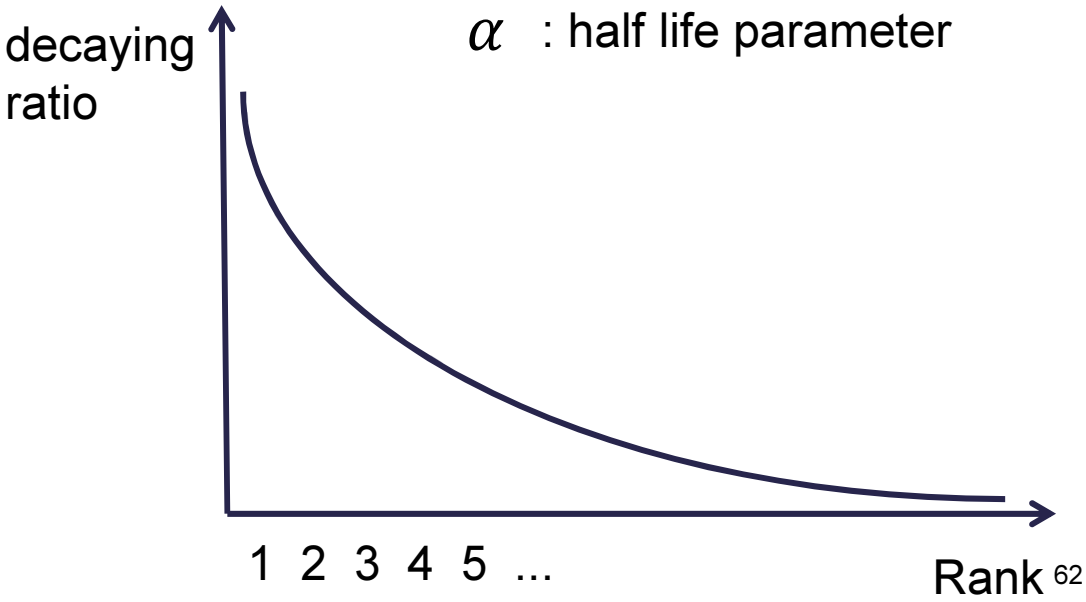
$$R_u = \sum_j \frac{\max(r_{u,j} - default, 0)}{2^{(j-1)/(\alpha-1)}}$$

$r_{u,i}$: user u's actual rating to item ranked at j

$default$: default rating value (usually average)

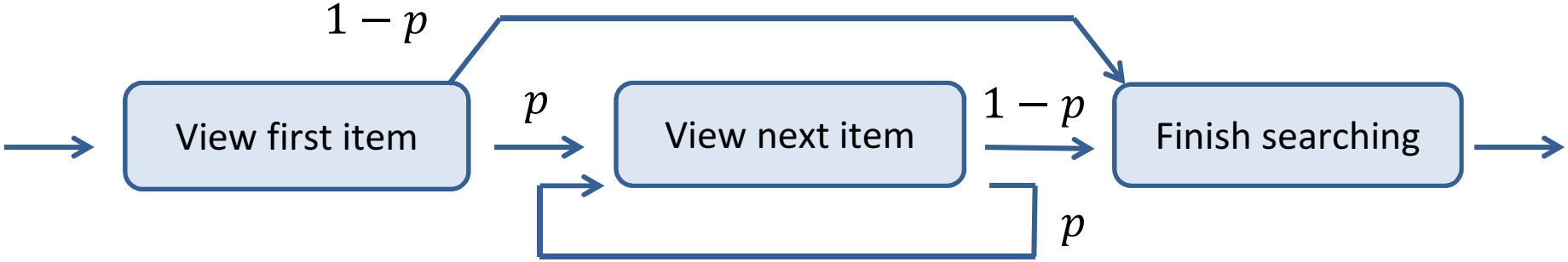
α : half life parameter

Rank	Item
1	Item A
2	Item B
3	Item C
4	Item D
5	Item E



RBP

- RBP (Rank biased prediction) [Moffat 08]
  - But, incorporate user models (when user stops browsing the recommendation list)



$$RBP = (1-p)\sum_{i=1}^{n} g_i \cdot p^{i-1}$$

gi: the degree of relevance of the item at rank i to the user's preference
p: parameter that models how persistent a user is while looking through the ranked list. (Usually estimated from click logs)

Cascade user model

- Cascade user model [Chapelle 09]
  - consider the relevance of items existing in the higher rank
    - come from the idea that once the user is satisfied with an item, he/she terminates the search and items below this result are not examined and clicked.

$$P(stop\ at\ rank\ r) = \prod_{i=1}^{r-1}(1 - R_i)R_r$$

Ri: Probability the user is satisfied and stops browsing ranking list. Values can be set as a function reflected from the relevance to the user preference.

ERR

- ERR (Expected reciprocal rank) [Chapelle 09]
  - One of cascade based metric
  - Take summation of the probability that the user stops examining the ranking in the r-th position.
  - The above probability is influenced by the upper ranking.

$$ERR = \sum_{r=1}^{n} \varphi(r) P(user\ stops\ at\ position\ r)$$

$$ERR = \sum_{r=1}^{n} \varphi(r) \prod_{i=1}^{r-1} (1 - R_i)\, R_r$$

$$\varphi(r) = \frac{1}{r} \quad \text{or} \quad \varphi(r) = \frac{1}{log_2(r+1)}$$

$$R_i = \frac{2^g - 1}{2^{g_{max}}}$$

g: relevance grade to the user preference
gmax: The maximum grade in the prefixed scale. (e.g. 5 in five scale (1-5))

Accuracy of list relevance and Accuracy based on ranking position

- Pros and Cons
  - [Pros] Evaluate the ranking list (Actually, users receive recommendation in a ranking list)
  - [Pros] Recent metrics consider the ranking position (Normal users do not browse the list to the tail)
  - [Cons] Do not evaluate the recommendation timing / context (when, where, how) [Olmo 08]

Discovery-oriented Metrics

Discovery-oriented metrics

- Why beyond accuracy
  - We should consider more perspectives rather than accuracy
- Problems of accuracy indices
  - Do not consider the system's utility
  - Users get tired of recommendation if it recommends similar items to the past.
  - User has already made a decision of purchase for known items.

Usefulness

- Usefulness: Whether the recommendation results provide utility to users. [Herlocker 04]
  - Utility metrics except discovery-oriented metrics
    - Measuring the system's utility
      - Coverage, Learning rate, Confidence, Trustworthiness
  - Discovery-oriented metrics
    - Measuring whether the recommendation is new to the user
      - Serendipity metrics
      - Novelty metrics
      - Diversity metrics

Utility metrics (except discovery-oriented metrics)

Coverage

- Coverage
  - Measure how many items the system can make a prediction.
  - Collaborative filtering cannot make a prediction to item which has no ratings from users.
  - Content-based filtering cannot make a prediction to item which lack some feature values.

Prediction coverage

$$Coverage = \frac{|C_i|}{|B_i|}$$

[Sarwar 98]

B_i : Item set

C_i : Item set the system can make a prediction

Coverage

- Coverage
  - Catalogue coverage [Ge 10]
    - Consider the types of items during a fixed time length

$$Catalogue\ coverage = \frac{\left|\bigcup_{j=1\cdots N} ⊇ L_j\right|}{|B|}$$

j : The number of times of the recommendation

L_j : j-th time recommendation list B : Item set

  - User coverage [Kawamae 10]

$$User\ coverage = \frac{|V_i|}{|U_i|}$$

U_i : User set

V_i : User set the system can recommend at least one item

Learning rate

- Learning rate
  - How fast the RS provides a well-adapted recommendation after changing the user's preference has changed
    - Related to cold-start problem [Schein 02]
    - User satisfaction decrease when the system's does not provide recommendation soon [Jones 07]
    - Time after the change of preference [Koychev 00]
    - Keep measuring the accuracy since the user has started the RS [Rashid 02]

Confidence

- Confidence
  - How confident that the system thinks for the recommendation [Sinha 01, Herlocker 04]
    - Calculated from the number of users or items which are used for creating the recommendation
    - Calculated from similarity of neighborhood in CF [Bell 07]

Trustworthiness

- User's trust to the RS
  - Should show the RS's ability to predict the user's preference and interest
  - Difficult to obtain it w/o showing users' well-known items [Sinha 01]
  - Users do not continue to use the RS unless they do not trust the system [Cramer 08]
  - Explanation to rec. results increase trust [Tintarev 07]
  - Obtained by direct questionnaire to users in online evaluation [Bonhard 07, Cramer 08]
  - Estimated from user's usage frequency [O'Donovan 05]

Metrics for discovery

Discovery

- Discovery: General notion regarding the system's ability to provide new and various types of items to the user
  - Serendipity: Whether the recommendation gives the surprise to the user. (The user cannot search the item by oneself.)
  - Novelty: Whether the recommended items are unknown to the user.
  - Diversity: Whether the system can recommend various types of items to users.

Discovery-oriented metrics

- Viewpoints
  - Evaluation target: List, user set, item set, item pair?
  - Required data (information): only rating matrix (user x item) , ontology (item category), other RS, other dataset (regarding the novelty, serendipity)

Discovery-oriented metrics: Diversity

Metrics	Type of Discovery	Target	Other Info.	Inventor
Aggregate diversity	Diversity	User set	None	Adomavicius, IEEE 2012
Inter-user diversity	Diversity	User set	None	Zhou, NAS 2010
List personalization metric	Diversity	List	None	Zhou, NAS 2010
Gini coefficient	Diversity	Item set	None	Fleder, EC 2007
Temporal diversity	Diversity	List pair	None	Lathia, SIGIR 2010
Intra-list similarity	Diversity	List	Ontology	Ziegler, WWW 2005
Subtopic retrieval	Diversity	List	Ontology	Zhai, SIGIR 2003
MMR	Diversity	List	Ontology	Carbonell, SIGIR 1998
α-nDCG	Diversity	List	Ontology	Clarke, SIGIR 2008

Discovery-oriented metrics: Novelty and Serendipity

Metrics	Type of Discovery	Target	Other Info.	Inventor
Discovery ratio	Novelty	List	Acquaintance	Hijikata, IUI 2009
Precision of novelty	Novelty	List	Acquaintance	Hijikata, IUI 2009
Item novelty	Novelty	Item	Ontology	Zhang, RecSys 2008
Temporal novelty	Novelty	List	None	Lathia, SIGIR 2010
Novelty based on HLU	Novelty	List	None	Shani, RecSys 2008
Long tail metric	Novelty	List	None	Celma, RecSys 2008
Generalized novelty model	Novelty	List	None/Ontology	Vargas, RecSys 2011
Unexpectedness	Serendipity	List	Other system	Murakami, LNCS 2008
Entropy-based diversity	Serendipity	List	Other systems	Bellogin, HetRec 2010
Unserendipity	Serendipity	List	Ontology	Zhang, WSDM 2012
HLU of serendipity	Serendipity	List	Serendipity rating	Murakami, JSAI 2009

Diversity metrics

Diversity: Aggregate diversity

- Aggregate diversity [Adomavicius 2012]
  - Aggregate the types of items recommended to all users
  - High value to this metric indicates that the system provides different items to users

$$Agg_{div} = \left| \bigcup_{u \in U} L_u \right|$$

U : User set
L_u : Recommendation list for User u

Diversity: Inter-user diversity

- Inter-user diversity
  - How the system provides different recommendation list among users [Zhou 10]

$$d_{u,v} = 1 - \frac{|\text{ב}L_u \cap \text{ב}L_v|}{N} \qquad N = |\text{ב}L_u| = |\text{ב}L_v|$$

Degree of personalization for the system

Inter-user diversity (IUD)

$$IUD = \frac{1}{N_{|U|}} \sum_{u,v \in U} d_{u,v}$$

U : User set

$N_{|U|}$: Num. of two pairs in U

L_u : Recommendation list for User u

Diversity: List personalization metric

- List personalization metric [Zhou 10]

  Degree of personalization for the list

  Probability of item b selected by a user

$$p_b = \frac{|U_b|}{|U|}$$

U_b : The number of users who selected item b

  Self entropy of item b

$$I_b = log_2\left(\frac{|U|}{|U_b|}\right)$$

  Degree of personalization of the list

$$Per(L_i) = \frac{\sum_{b_j \in \text{ב} L_i} log \frac{|U|}{|U_b|}}{|\text{ב} L_i|}$$

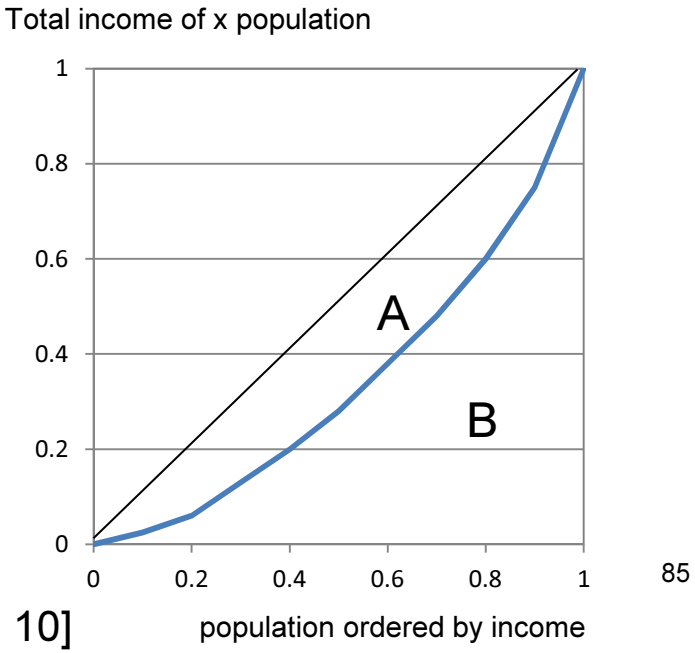
Diversity: Gini coefficient

- Gini coefficient
  - x-axis: x% of the population ordered by income
  - y-axis: Total income of the population cumulatively earned by x population

$$G = B/(A+B)$$

When applying to recommendation results

x-axis: x% of the items ordered by the frequency in the lists
y-axis: Total frequency of x% of items in the list



Applied for measuring the diversity
to e-commerce site of music [Fleder 07, Kawamae 10]

Diversity: Temporal diversity

- Temporal diversity [Lathia 10]
  - Measure the system outputs different items when the time is different (t1 and t2)

$$div_{tmp}(L_1, L_2, N) = \frac{|\{x \in L_2 | x! \in L_1\}|}{N} \qquad N = |\text{ב}L_1| = |\text{ב}L_2|$$

Diversity: Intra-list similarity

- Diversity according to content [Ziegler 05]
  - Measure the diversity of recommendation list
  - Calculated using the similarity of two items
  - Feature vectors or categories are usually used.

$$Diversity = \sum_{l \in \supset L_i, m \in \supset L_i} \frac{1}{similarity(l, m)}$$

- Intra-list similarity (Diversity) [Ziegler 05]

$$ILS(L_i) = \frac{\sum_{b_k \in L_i} \sum_{b_e \in L_i \, b_k \neq b_e} similarity(b_k, b_e)}{|L_i| C_2}$$

Diversity: Subtopic retrieval metric

- Subtopic retrieval metric [Zhai 03]

  Measure how many topics (categories) are covered by recommendation list among the total number of topics

$$S_{recall} = \frac{\left|\bigcup_{i=1}^{K} subtopics(s_i)\right|}{n}$$

K : The length of the recommendation list

s_i : Item in the recommendation list

n : The total number of topics

Diversity: MMR

- MMR (Maximal marginal relevance) of the list [Carbonell 98]
  - MMR is originally the item selection method for selecting a document to the search result
  - This idea can be used for evaluating the search result list (recommendation list)

$$MMR_{List} = \sum_{d_i \in L} \left\{ \alpha rel(d_i) - (1-\alpha) max_{d_j \in L^{j-1}} sim(d_i, d_j) \right\}$$

L : Recommendation list

d_i : Item in the list

L^{j-1} : Recommendation list until jth rank

rel : Whether the user likes the item

Diversity: α - nDCG

- Diversity (α -nDCG) [Clarke 08]
  - Originally developed in IR field
  - Should reduce redundancy and cover wide categories (‘nugget’ in original)
  - Should cover wide categories in recommendation

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{log_2(i)}$$

$$DCG_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{log_2(i+1)}$$

$$nDCG_p = \frac{DCG_p}{IDCG_p}$$

Diversity: α - nDCG

- Diversity (α-nDCG)

  for calculating rel_k

$$rel_k = \sum_{i=1}^{m} J(d_k, i)(1-\alpha)^{r_{i,k-1}}$$

  i: i-th category (nugget)
  dk: item (document) ranked in k-th position)
  J(dk,i): whether dk includes nugget i (0 or 1)
  ri,k-1: the number of d including nugget i

Novelty metrics

Novelty metrics

- How to measure novelty
  - Ask user about their acquaintance to a specific item [Hijikata 09, Celma 08]
  - Calculate the general popularity or similarity among items [Celma 08, Shani 08, Meyer 12]

Input data set (for novelty evaluation)

[Hijikata 09]

- Rating matrix

$$
u \begin{pmatrix}
5 & 1 & 0 & 0 \\
0 & 4 & r & 0 \\
0 & 6 & 2 & 2 \\
6 & 0 & 0 & 7
\end{pmatrix}
\quad i
$$

Learning set | Test set

- Acquaintance matrix

$$
u \begin{pmatrix}
1 & 1 & 0 & - \\
- & 1 & r & 0 \\
0 & 1 & 1 & 1 \\
1 & - & 0 & 1
\end{pmatrix}
\quad i
$$

Learning set | Test set

Novelty: Discovery ratio

- Discovery ratio [Hijikata 09]
  - How many unknown items are recommended in the list?

$$Discovery = \frac{|D_i \cap ב L_i|}{|ב L_i|}$$

D_i : User i's unknown item set in the test set L_i : System's recommendation list

Novelty: Precision of novelty

- Precision of Novelty [Hijikata 09]
  - Whether the item is unknown to the user and the user will like the item.

$$Precision(novelty) = \frac{|C_i \cap \text{ב} L_i|}{|\text{ב} L_i|}$$

$$Recall(novelty) = \frac{|C_i \cap \text{ב} L_i|}{|C_i|}$$

C_i : User i's unknown and favorite item set in the test set

L_i : System's recommendation list

Novelty: Item novelty

- Item novelty [Zhang 08]
  - Measure the novelty of recommended item
    - By using the intra-list diversity [Ziegler 05]

$$Nov_{item}(i) = N\{Div(\text{ב}L) - Div(\text{ב}L - \{i\})\}$$

$$= \frac{1}{N-1}\sum_{j \in \text{ב}L} d(i,j)$$

$|\text{ב}L| = N$ $d(i,j)$: distance function between items

$Div\ ()$: diversity function of the list

Novelty: Temporal novelty

- Temporal novelty [Lathia 10]
  - Measure whether the system outputs different items from past recommended items

$$Nov_{tmp}(L_i) = \frac{\left|\{x \in L_i | x! \in S_{past}\}\right|}{|L_i|} \qquad S_{past} = \bigcup_{j=1}^{i-1} L_i$$

Novelty: Novelty based on HLU

- Novelty based on Half-life utility [Shani 08]

Half-life utility metric

$$R_u = \sum_j \frac{\max(r_{u,j} - default, 0)}{2^{(j-1)/(\alpha-1)}}$$

[Breese, UAI'98]

$r_{u,i}$: user u's actual rating to item ranked at j

$default$: default rating value (usually average)

α : half life parameter

Introducing the general popularity of items

$$f(i) = log_2\left(\frac{n}{n_i}\right) \qquad NHLU_u = \sum_i^N f(i) \frac{\max(rel_{u,i} - d, 0)}{2^{(i-1)/(\alpha-1)}}$$

n : Number of users n_i : Num. of users selecting item i

Novelty: Metrics based on long tail

- Metrics based on long tail [Celma 08]
  - Dividing the item to three categories according to its popularity: HEAD, MID TAIL
  - Extract item a_i and a_j in top N recommendation.
  - Evaluate the ability to recommend novel items by the following evaluation matrix.

$a_i \longrightarrow a_j$	HEAD	MID	TAIL
HEAD	45.32%	54.68%	0.00%
MID	5.43%	71.75%	22.82%
TAIL	0.24%	17.16%	82.60%

If HEAD->HEAD is large, novelty is low.

Values are reprinted from [Celma 08]

Novelty: General model

- General model for novelty [Vargas 11]

$$nov(L|\theta) = C \sum_{i \in L} \underbrace{p(choose|i,u,L)}_{\text{browsing model}} \underbrace{nov(i|\theta)}_{\text{item novelty model}}$$

L : Recommendation list θ: Definition of novelty

  - Item novelty model
    - Popularity based item novelty

$$nov(i|\theta) = -log_2 p(seen|i,\theta)$$

θ : How frequently the item is selected by people

    - Distance based item novelty

$$nov(i|\theta) = \sum_{j \in \theta} p(j|choose,\theta,i)d(i,j)$$

θ : The user's previously selected items $d(i,j)$: Distance function

Novelty: General model

- Browsing model

$$p(choose|i,u,L) = p(seen|i,u,L)p(rel|i,u)$$

The user does not select unseen items or disliked items

$$p(seen|i_k,u,L) = \prod_{l=1}^{k-1} p(cont|l,u,L)$$

The user does not see item ik if he stops browsing until k-1 rank

In simple, $p(cont|l,u,R) = p_0$

If using the idea of ERR $p(seen|i_k,u,L) = \prod_{l=1}^{k-1}(1 - p(rel|i_l,u))$

$p(rel|i,u)$: Calculated from the correct data

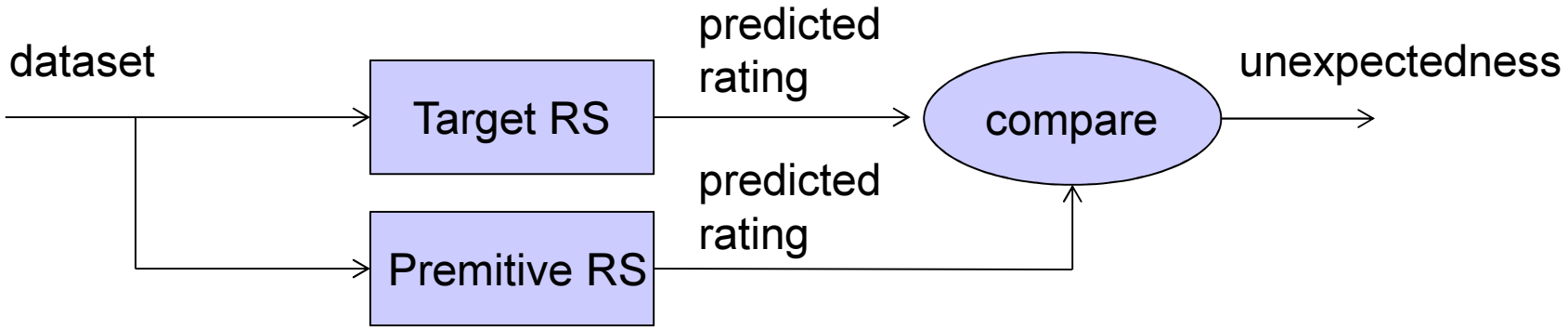
Serendipity metrics

Serendipity metrics

- How to measure serendipity
  - Ask user about their surprise to the recommendation of a specific item [Murakami 09]
  - Calculate the expectation difficulty by using other systems [Murakami 08, Ge 10, Bellogin 10]

Serendipity: Unexpectedness

- Unexpectedness [Murakami 08]



$$EXEXP = \frac{1}{|\text{ב}L_i|} \sum_{i=1}^{|\text{ב}L_i|} \max(P(s_i) - prim(s_i), 0) \cdot rel(s_i)$$

s_i : Item at the i-th rank

L_i : Recommendation list to be evaluated

$rel(s_i)$: User's rating to item si

$P(s_i)$: The predicted rating by the primitive system

$prim(s_i)$: The predicted rating by the target system

Serendipity: Unexpectedness

- Unexpectedness [Ge 10, Adamopoulos 13]

$$UNEXP = RS / PM$$

$$SRDP = \frac{|UNEXP \cap USEFUL|}{N}$$

RS : Item set output by RS

PM: Item set output by primitive System [Ge 10]

PM: Item set similar to previously seen items [Adamopoulus 10]

$USEFUL$: Useful Item set

Serendipity: Entropy-based diversity

- Entropy-based diversity [Bellogin 10]
  - Checks whether an item in the recommendation list of one RS is also recommended by other RS.

$$div_{a,u} = -\sum_{i \in \text{ב} L_{a,u} \cap R_u} p_{u,i} \cdot log p_{u,i} \qquad p_{u,i} = \frac{\sum_{a \in A} \delta(a,u,i)}{|A|}$$

A : RS set a : Target system to evaluate

$L_{a,u}$: Recommendation list provided by system a for user u

R_u : Relevant item to user u

δ : 1 if $i \in \text{ב} L_{a,u} \cap R_u$ and 0 otherwise

Serendipity: Unserendipity metric

- Unserendipity [Zhang 12]
  - Calculate item's unserendipity by measuring the similarity to items in the user's history

$$Unseren = \sum_{u \in U} \frac{1}{|U||H_u|} \sum_{h \in H_u} \sum_{i \in \supset L_u} \frac{sim(i,h)}{|\supset L_u|}$$

U : User set H_u : User history L_u : Recommendation list

$sim()$: Similarity between items

Input data set (for serendipity evaluation)

- Serendipity
  - Whether the item is surprisingly found and the user will like the item.
  - It is difficult to define formally in formula.
  - Surprise is difficult to be detected or measured.

Serendipity: Half-life utility of serendipity

- Half-life utility of serendipity [Murakami 09]

$$HLT_{serendipity} = \frac{R^{ser}}{R_{max}}$$

$$R^{ser} = \sum_{i=1, s_i \in J \cap Ser}^{N} \frac{1}{2^{(i-1)(\beta-1)}} \qquad R_{max} = \sum_{i=1, s_i \in Ser}^{M} \frac{1}{2^{(i-1)(\beta-1)}}$$

M : The number of items in the dataset

Ser : The user's serendipitous items in the dataset

J : Top N items in the recommendation results

Challenge of improving discovery

- Recommending only novel items decrease the user's satisfaction
  - Users generally prefer known items [Sinha 01]
  - Overall evaluation shows that novelty decreases the user satisfaction [Ekstrand 14]
- Take care when recommending novel items
  - Consider user's experience in RS
  - Explain novel item recommendation in advance
  - Explain each recommended novel items

Conclusion

- Almost all the evaluation metrics for offline test is introduced in this survey.
- Accuracy metrics are categorized to accuracy of estimated rating, estimated ranking, list relevance, and accuracy based on raking positon.
  - Most of the accuracy metrics are presented in well-defined formula
- Discovery-oriented metrics are categorized to evaluating the diversity, novelty and serendipity
  - Many challenges exist in discovery-oriented metrics for the tradeoff between accuracy and discovery.

Reference (1)

[Adamopoulos 13] Adamopoulos, P. and Tuzhilin, A.: On Unexpectedness in Recommender Systems: Or How to Better Expect the Unexpected, ACM Transactions on Intelligent Systems and Technology, Vol. 1, No. 1, pp. 1-51 (2013)

[Adomavicius 05] Adomavicius, G. and Tuzhilin, A.: Toward the Next Generation of Recommender Systems: a Survey of the State-of-the-art and Possible Extensions, IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 6, pp. 734-749 (2005)

[Adomavicius 07] Adomavicius, G. and Kwon, Y.: New Recommendation Techniques for Multicriteria Rating Systems, IEEE Intelligent Systems, Vol. 22, No. 3, pp. 48-55 (2007)

[Adomavicius 12] Adomavicius, G. and Kwon, Y.: Improving Aggregate Recommendation Diversity Using Ranking-Based Techniques, IEEE Transactions on Knowledge and Data Engineering, Vol. 24, No. 5, pp. 896-911 (2012)

[Amatriain 09] Amatriain, X., Pujol, J. M., and Oliver, N.: I Like It... I Like It Not: Evaluating User Ratings Noise in Recommender Systems, User Modeling, Adaptation, and Personalization (Lecture Notes in Computer Science), Vol. 5535, pp. 247-258 (2009)

Reference (2)

[Bell 07] Bell, R. M., Koren, Y., and Volinsky, C.: Modeling Relationships at Multiple Scales to Improve Accuracy of Large Recommender Systems, in Proc. of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (ACM KDD'07), pp. 95-104 (2007)

[Bellogin 10] Bellogin, A., Cantador, I., and Castells, P.: A study of Heterogeneity in Recommendations for a Social Music Service, in Proc. of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec'10), pp. 1-8 (2010)

[Bollen 10] Bollen, D., Knijnenburg, B. P., Willemsen, M. C., and Graus, M.: Understanding Choice Overload in Recommender Systems, in Proc. of the 2010 ACM Conference on Recommender Systems (ACM RecSys'10), pp. 63-70 (2010)

[Bonhard 07] Bonhard, P., Harries, C., McCarthy, J., and Sasse, M. A.: Accounting for Taste: Using Profile Similarity to Improve Recommender Systems, in Proc. of the SIGCHI Conference on Human Factors in Computing Systems (ACM CHI'07), pp. 1057-1066 (2007)

[Breese 98] Breese, J. S., Heckerman, D., and Kadie, C.: Empirical Analysis of Predictive Algorithms for Collaborative Filtering, in Proc. of Fourteenth Conference on Uncertainty in Artificial Intelligence (UAI'98), pp. 43-52 (1998)

Reference (3)

[Buckley 05] Buckley, C. and Voorhees, E. M.: Retrieval System Evaluation in TREC: Experiment and Evaluation in Information Retrieval, MIT Press (2005)

[Carbonell 98] Carbonell, J. and Goldstein, J.: The Use of MMR, Diversity-based Reranking for Reordering Documents and Producing Summaries, in Proc. of 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (ACM SIGIR'98), pp. 335-336 (1998)

[Celma 08] Celma, O. and Herrera, P.: A New Approach to Evaluating Novel Recommendations, in Proc. of the 2008 ACM Conference on Recommender Systems (ACM RecSys'08), pp. 179-186 (2008)

[Chapelle 09] Chapelle, O., Metlzer, D., Zhang, Y., and Grinspan, P.: Expected Reciprocal Rank for Graded Relevance, in Proc. of the 18th ACM Conference on Information and Knowledge Management (ACM CIKM'09), pp. 621-630 (2009)

[Clarke 08] Clarke, C. L., Kolla, M., Cormack, G. V., Vechtomova, O., Ashkan, A., Buttcher, S., and MacKinnon, I.: Novelty and Diversity in Information Retrieval Evaluation, in Proc. of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (ACM SIGIR'08), pp. 659-666 (2008)

Reference (4)

[Cosley 03] Cosley, D., Lam, S. K., Albert, I., Konstan, J. A., and Riedl, J.: Is Seeing Believing - How Recommender Interfaces Affect Users' Opnions, in Proc. of the Conf. on Human Factors in Computing Systems (ACM CHI'03), pp. 585-592 (2003)

[Cramer 08] Cramer, H., Evers, V., Ramlal, S., Someren, van M., Rutledge, L., Stash, N., Aroyo, L., and Wielinga, B.: The Effects of Transparency on Trust in and Acceptance of a Content-based Art Recommender, User Modeling and User-Adapted Interaction, Vol. 18, No. 5, pp. 455-496 (2008)

[Craswell 08] Craswell, N., Zoeter, O., Taylor, M., and Ramsey, B.: An Experimental Comparison of Click Position-bias Models, in Proc. of International Conference on Web Search and Data Mining (ACM WSDM'08), pp. 87-94 (2008)

[Deshpande 04] Deshpande, M. and Karypis, G.: Item-based Top-N Recommendation Algorithms, ACM Transactions on Information Systems (TOIS), Vol. 22, No. 1, pp. 143-177 (2004)

[Ekstrand 14] Ekstrand, M., Harper, F. M., Willemsen, M., and Konstan, J.: User Perception of Differences in Movie Recommendation Algorithms, in Proc. of the fourth ACM Conference on Recommender Systems (ACM RecSys'10) (2014)

Reference (5)

[Fleder 10] Fleder, D. M. and Hosanagar, K.: Recommender Systems and their Impact on Sales Diversity, in Proc. of the 8th ACM conference on Electronic Commerce (EC'10), pp. 192-199 (2010)

[Garner 60] Garner, W. R.: Rating Scales, Discriminability, and Information Transmission, Psychological Review, pp. 343-352 (1960)

[Ge 10] Ge, M., Delgado-Battenfeld, C., and Jannach, D.: Beyond Accuracy: Evaluating Recommender Systems by Coverage and Serendipity, in Proc. of the of the fourth ACM Conference on Recommender Systems (RecSys'10), pp. 257-260 (2010)

[Gunawardana 09] Gunawardana, A. and Shani, G.: A Survey of Accuracy Evaluation Metrics of Recommendation Tasks, The Journal of Machine Learning Research, Vol. 10, pp. 2935-2962 (2009)

[Herlocker 04] Herlocker, J. L., Konstan, J. A., Terveen, L. G., and Riedl, J. T.: Evaluating Collaborative Filtering Recommender Systems, ACM Transactions on Information Systems (TOIS), Vol. 22, No. 1, pp. 5-53 (2004)

Reference (6)

[Hijikata 09] Hijikata, Y., Shimizu, T., and Nishida, S.: Discovery-oriented Collaborative Filtering for Improving User Satisfaction, in Proc. of the International Conference on Intelligent User Interfaces (ACM IUI'09), pp. 67-76 (2009)

[Hijikata 12] Hijikata, Y., Kai, Y., and Nishida, S.: The Relation between User Intervention and User Satisfaction for Information Recommendation, in Proc. of the 27th Annual ACM Symposium on Applied Computing (ACM SAC 2012), pp. 2002-2007 (2012)

[Hill 95] Hill, W., Stead, L., Rosenstein, M., and Furnas, G.: Recommending and Evaluating Choices in a Virtual Community of Use, in Proc. of the Conf. on Human Factors in Computing Systems (ACM CHI'95), pp. 194-201 (1995)

[Hosanagar 05] Hosanagar, K.: A Utility Theoretic Approach to Determining Optimal Wait Times in Distributed Information Retrieval, in Proc. of 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (ACM SIGIR'05), pp. 91-97 (2005)

[Jarvelin 02] Jarvelin, K. and Kekalainen, J.: Cumulated Gain-based Evaluation of IR Techniques, ACM Transactions on Information Systems (TOIS), Vol. 20, No. 4, pp. 422-446 (2002)

Reference (7)

[Jones 07] Jones, N. and Pu, P.: User Technology Adoption Issues in Recommender Systems, in Proc. of the 2007 Networking and Electronic Commerce Research Conference (NAEC'07), pp. 379-394 (2007)

[Kamishima07] Kamishima, T.: Algorithms of Recommender Systems (1), Journal of JSAI (The Japanese Society for Artificial Intelligence), Vol. 22, No. 6, pp. 826-837 (2007)

[Kawamae 10] Kawamae, N.: Serendipitous Recommendations via Innovators, in Proc. of the 33rd International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR'10), pp. 218-225 (2010)

[Koychev 00] Koychev, I. and Schwab, I.: Adaptation to Drifting User's Interests, in Proc. of ECML2000 Workshop: Machine Learning in New Information Age, pp. 39-46 (2000)

[Lam 06] Lam, S. K., Frankowski, D., and Riedl, J.: Do You Trust Your Recommendations? An Exploration Of Security and Privacy Issues in Recommender Systems, Emerging Trends in Information and Communication Security: Lecture Notes in Computer Science, Vol. 3995, pp. 14-29 (2006)

Reference (8)

[Lathia 10] Lathia, N., Hailes, S., Capra, L., and Amatriain, X.: Temporal Diversity in Recommender Systems, in Proc. of the 33rd international ACM SIGIR conference on Research and development in information retrieval (ACM SIGIR'10), pp. 210-217 (2010)

[Lops 11] Lops, P., Gemmis, de M., and Semeraro, G.: Content-based Recommender Systems: State of the Art and Trends, Recommender Systems Handbook, pp. 73-105 (2011)

[Manning 08] Manning, C. D., Raghavan, P., and Sch¨utze, H.: Introduction to Information Retrieval, Cambridge University Press (2008)

[Meyer 12] Meyer, F., Fessant, F., Clerot, F., and Gaussier, E.: Toward a New Protocol to Evaluate Recommender Systems, in Proc. of Workshop on Recommendation Utility Evaluation: Beyond RMSE (RUE 2012), pp. 9-14 (2012)

[Moffat 08] Moffat, A. and Zobel, J.: Rank-biased Precision for Measurement of Retrieval Effectiveness, ACM Transactions on Information Systems (TOIS), Vol. 27, No. 1 (2008)

Reference (9)

[Murakami 08] Murakami, T., Mori, K., and Orihara, R.: Metrics for Evaluating the Serendipity of Recommendation Lists, New Frontiers in Artificial Intelligence, Lecture Notes in Computer Science, Springer, Vol. 4914, pp. 40-46 (2008)

[Murakami 09] Murakami, T., Mori, T., and Orihara, R.: A Method to Enhance Serendipity in Recommendation and its Evaluation, Transaction of the Japanese Society for Artificial Intelligence, Vol. 24, No. 5, pp. 428-436 (2009)

[O'Donovan 05] O'Donovan, J. and Smyth, B.: Trust in Recommender Systems, in Proc. of the 10th International Conference on Intelligent User Interfaces (ACM IUI'05), pp. 167-174 (2005)

[Olmo 08] Olmo, del F. H. and Gaudioso, E.: Evaluation of Recommender Systems: A New Approach, Expert Systems with Applications, Vol. 35, pp. 790-804 (2008)

[Rashid 02] Rashid, A. M., Albert, I., Cosley, D., Lam, S. K., McNee, S. M., Konstan, J. A., and Riedl, J.: Getting to Know You: Learning New User Preferences in Recommender Systems, in Proc. of the 7th International Conference on Intelligent User Interfaces (ACM IUI'02), pp. 127-134 (2002)

Reference (10)

[Richardson 07] Richardson, M., Dominowska, E., and Ragno, R.: Predicting Clicks: Estimating the Click-through Rate for New Ads, in Proc. of the 16th International Conference on World Wide Web (ACM WWW'07), pp. 521-530 (2007)

[Robertson 97] Robertson, S. E.: The Probability Ranking Principle in IR, Journal of Documentation, Vol. 33, pp. 294-304 (1997)

[Sarwar 98] Sarwar, B. M., Konstan, J. A., Borchers, A., Herlocker, J., Miller, B., and Riedl, J.: Using Filtering Agents to Improve Prediction Quality in the GroupLens Research Collaborative Filtering System, in Proc. of the 1998 ACM Conference on Computer Supported Cooperative Work (ACM CSCW'98), pp. 345-354 (1998)

[Sarwar 00] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J.: Analysis of Recommendation Algorithms for e-commerce, in Proc. of the 2nd ACM Conference on Electronic Commerce (ACM EC'00), pp. 158-167 (2000)

[Schein 02] Schein, A. I., Popescul, A., Ungar, L. H., and Pennock, D. M.: Methods and Metrics for Cold-start Recommendations, in Proc. of of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (ACM SIGIR'02), pp. 253-260 (2002)

Reference (11)

[Shani 08] Shani, G., Chickering, M., and Meek, C.: Mining Recommendations From TheWeb, in Proc. of the 2008 ACM Conference on Recommender Systems (ACM RecSys'08), pp. 35-42 (2008)

[Stone 74] Stone, M.: Cross-Validatory Choice and Assessment of Statistical Predictions, Journal of the Royal Statistical Society B, Vol. 36, No. 1, pp. 111-147 (1974)

[Tintarev 07] Tintarev, N. and Masthoff, J.: A Survey of Explanations in Recommender Systems, in Proc. of the 2007 IEEE 23rd International Conference on Data Engineering Workshop, pp. 801-810 (2007)

[Vargas 11] Vargas, S. and Castells, P.: Rank and Relevance in Novelty and Diversity Metrics for Recommender Systems, in Proc. of the fifth ACM Conference on Recommender Systems (ACM RecSys'11), pp. 109-116 (2011)

[Yao 95] Yao, Y.: Measuring Retrieval Effectiveness Based on User Preference of Documents, Journal of the American Society for Information Science, Vol. 46, pp. 133-145 (1995)

Reference (12)

[Zhai 03] Zhai, C. X., Cohen, W. W., and Lafferty, J.: Beyond Independent Relevance: Methods and Evaluation Metrics for Subtopic Retrieval, in Proc. of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval (ACM SIGIR'03), pp. 10-17 (2003)

[Zhang 08] Zhang, M. and Hurley, N.: Avoiding Monotony: Improving the Diversity of Recommendation Lists, in Proc. of the second ACM Conference on Recommender Systems (ACM RecSys'08), pp. 123-130 (2008)

[Zhang 12] Zhang, Y. C., Se´aghdha, D. O´ ., Quercia, D., and Jambor, T.: Auralist: Introducing Serendipity into Music Recommendation, in Proc. of the fifth ACM international conference on Web search and data mining (WSDM'12), pp. 13-22 (2012)

[Zhou 10] Zhou, T., Kuscsik, Z., Liu, J.-G., Medo, M., Wakeling, J. R., and Zhang, Y.-C.: Solving the Apparent Diversityaccuracy Dilemma of Recommender Systems, in Proc. of the National Academy of Sciences, pp. 4511-4515 (2010)

[Ziegler 05] Ziegler, C.-N., McNee, S. M., Konstan, J. A., and Lausen, G.: Impro ving Recommendation Lists through Topic Diversification, in Proc. of the 14th International Conference on World Wide Web (WWW'05), pp. 22-32 (2005)

Contact

Osaka University
Associate Professor

Yoshinori Hijikata, Ph.D.

hijikata@sys.es.osaka-u.ac.jp

http://soc-research.org

Published online on Nov. 7th, 2014.