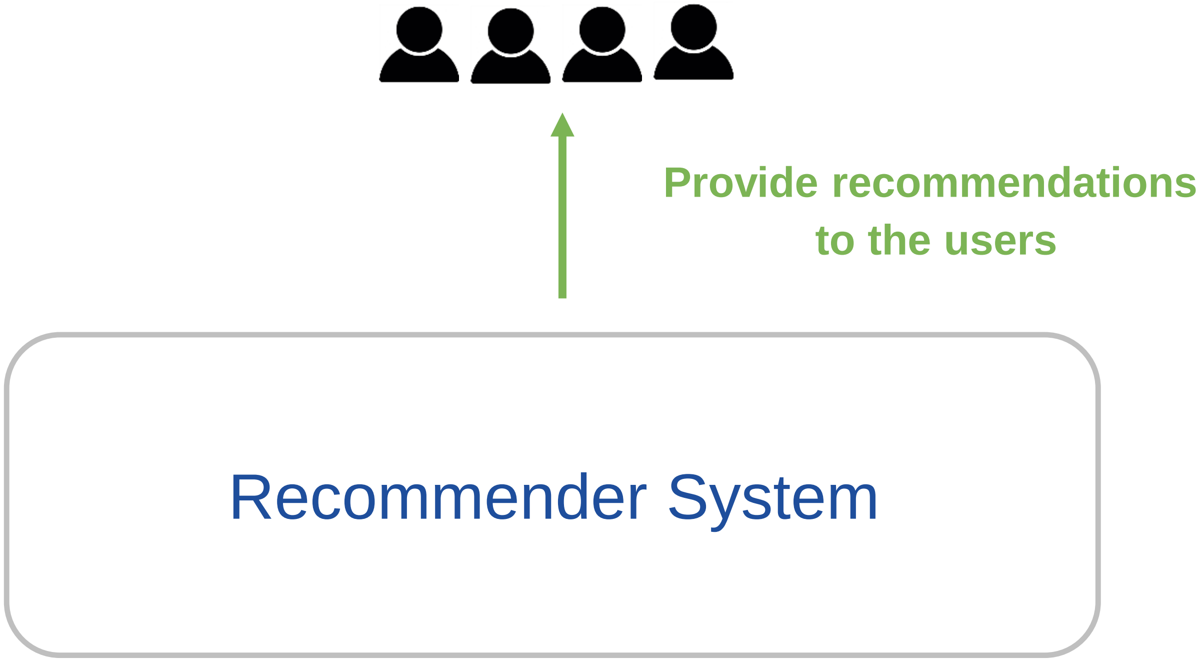
Introduction to Recommender Systems

Recommender Systems

Recommender System

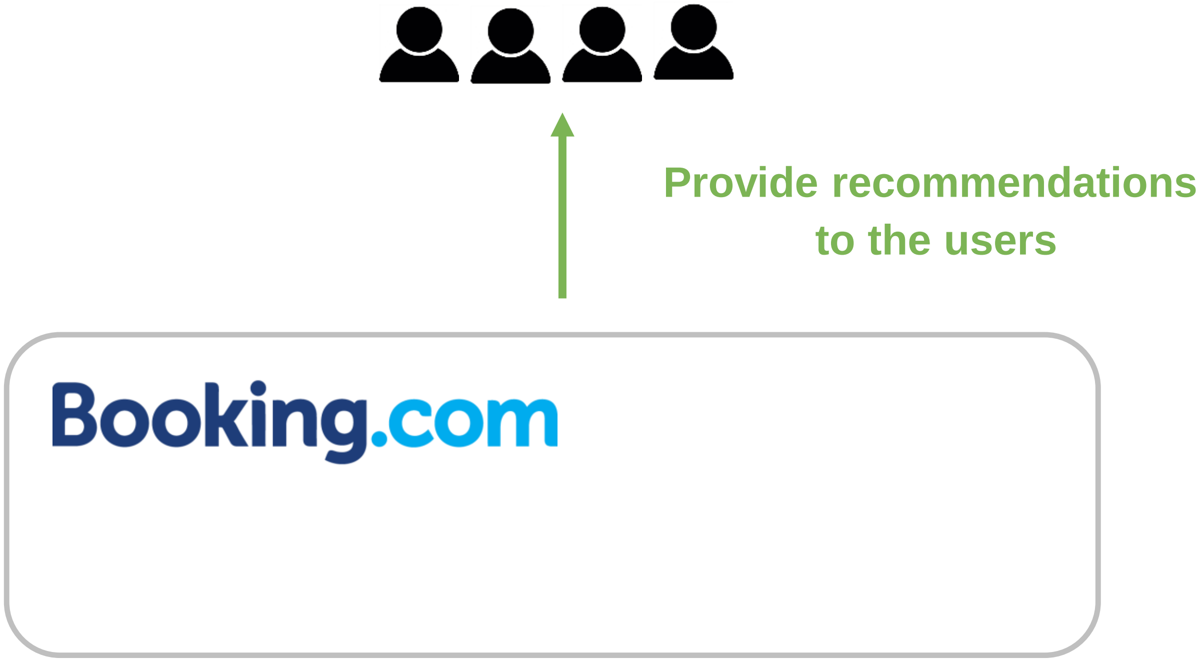
Recommender Systems



Provide recommendations to the users

Recommender System

Recommender Systems

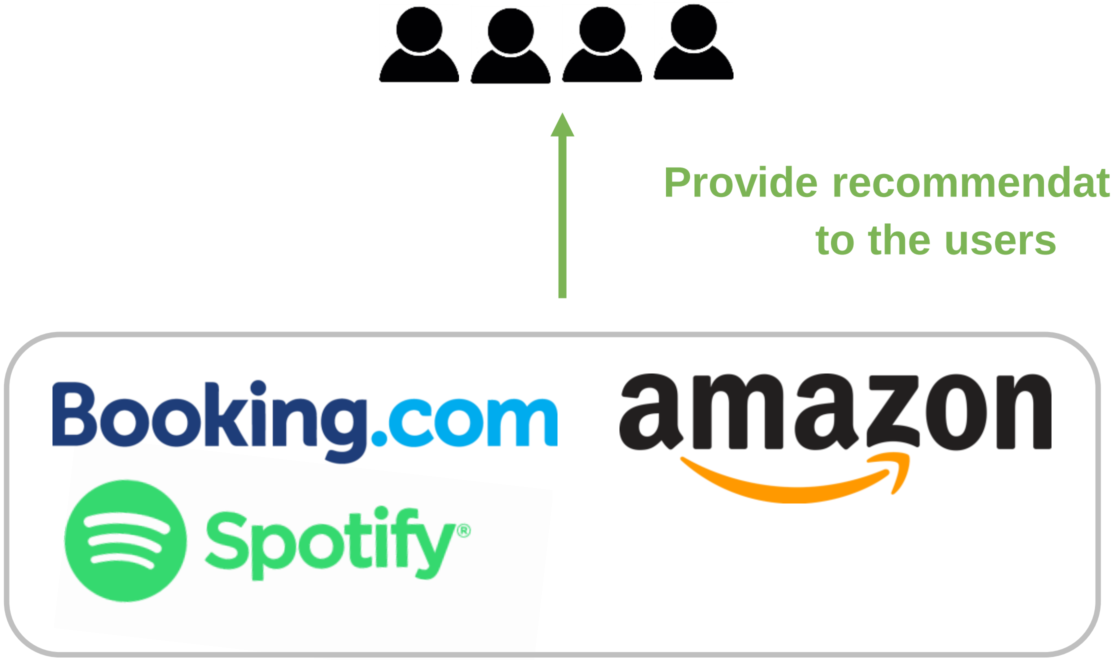


Provide recommendations to the users

Recommender Systems

Provide recommendations to the users

Recommender Systems

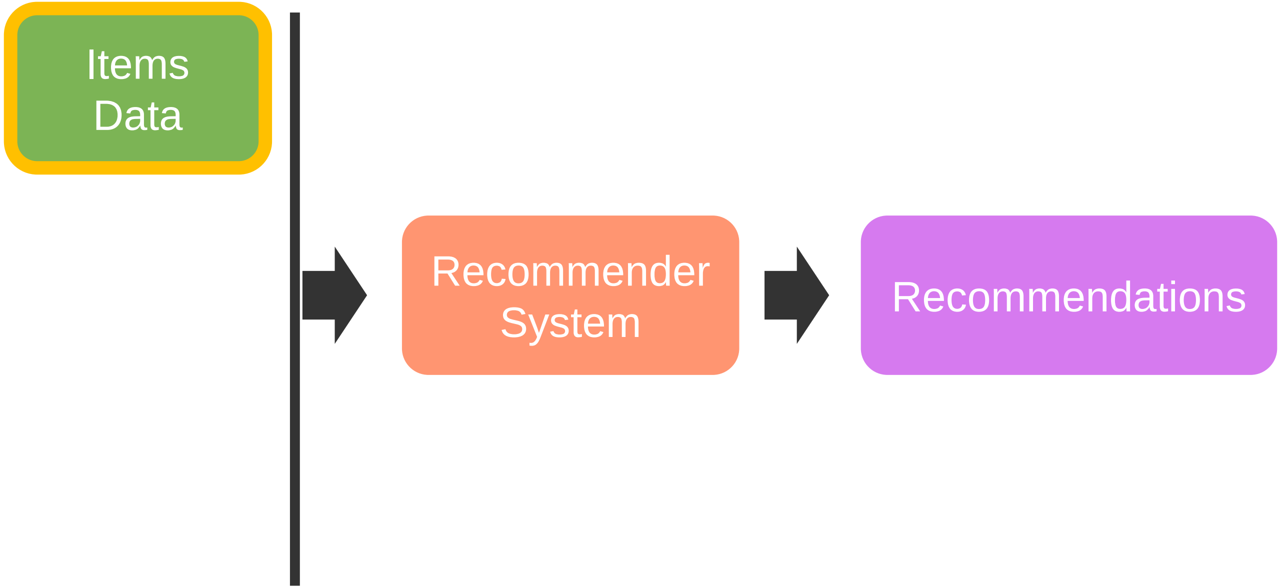


Provide recommendations to the users

Recommender Systems

Provide recommendations to the users

Input Data

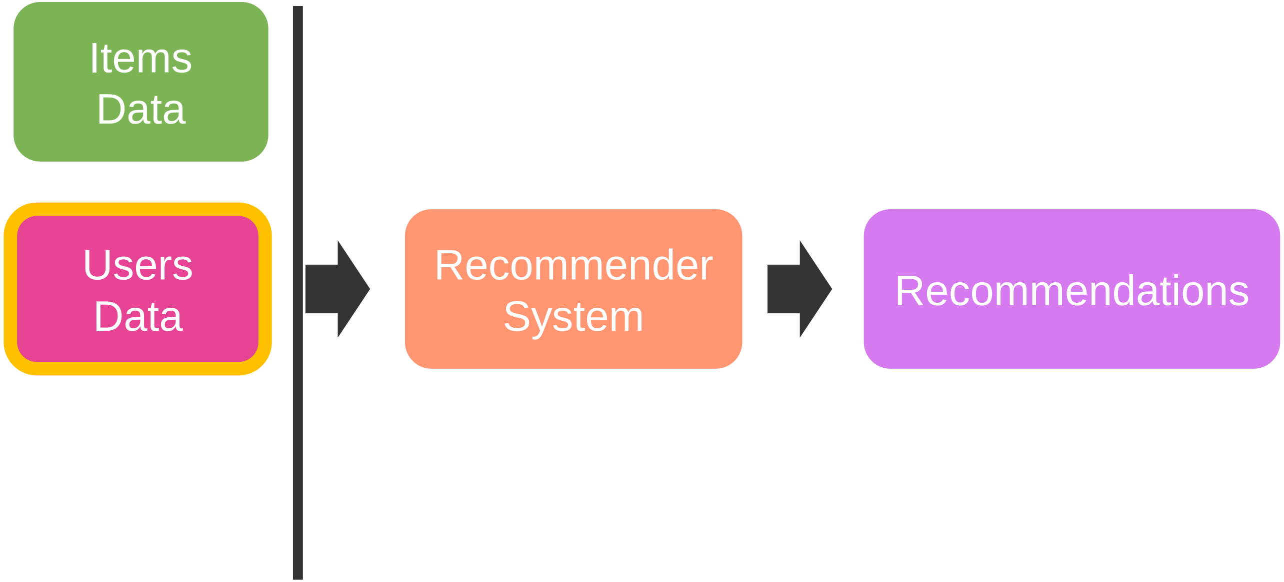


Items Data

Recommender System

Recommendations

Input Data



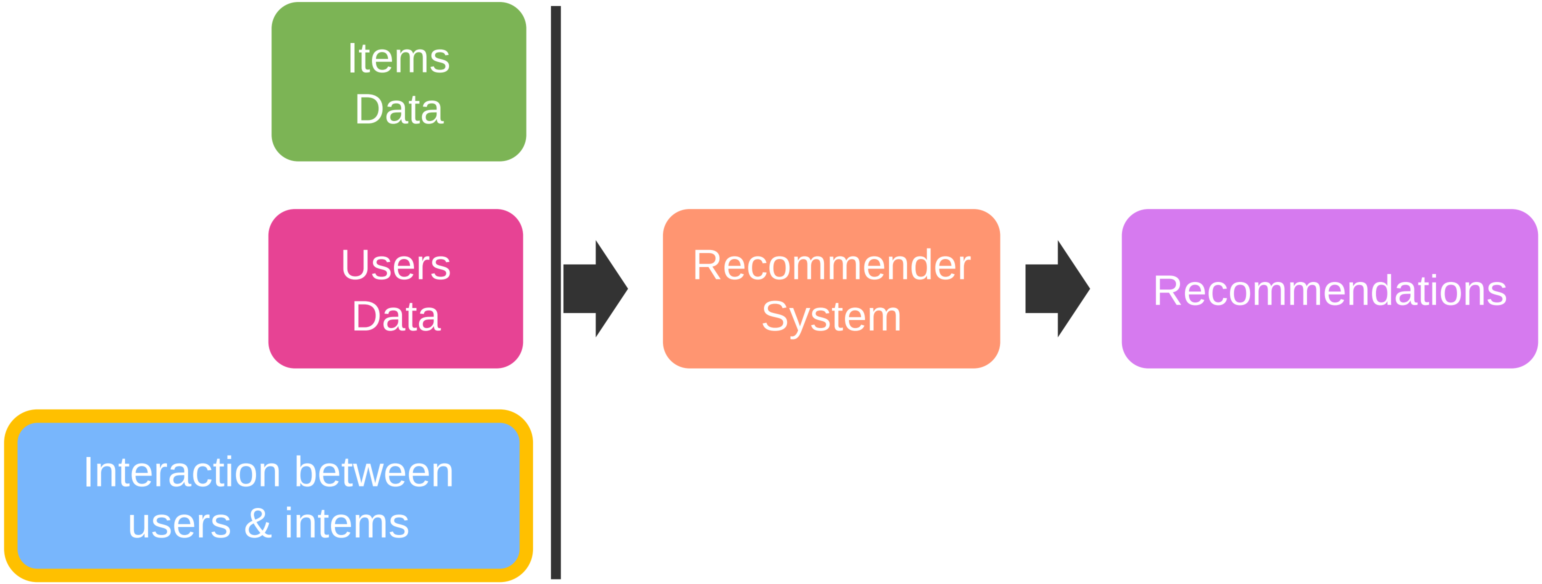
Items Data

Users Data

Recommender System

Recommendations

Input Data



Items Data

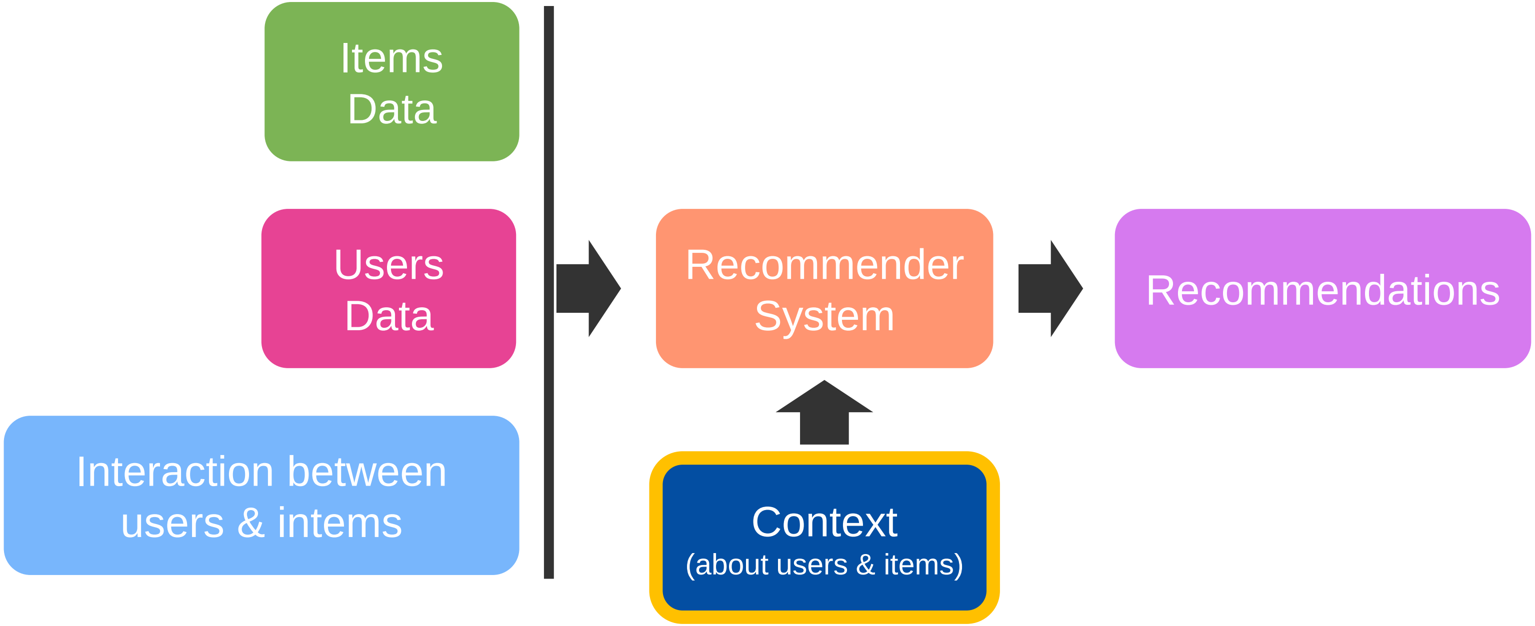
Users Data

Interaction between users & intems

Recommender System

Recommendations

Input Data



Items Data

Users Data

Interaction between users & intems

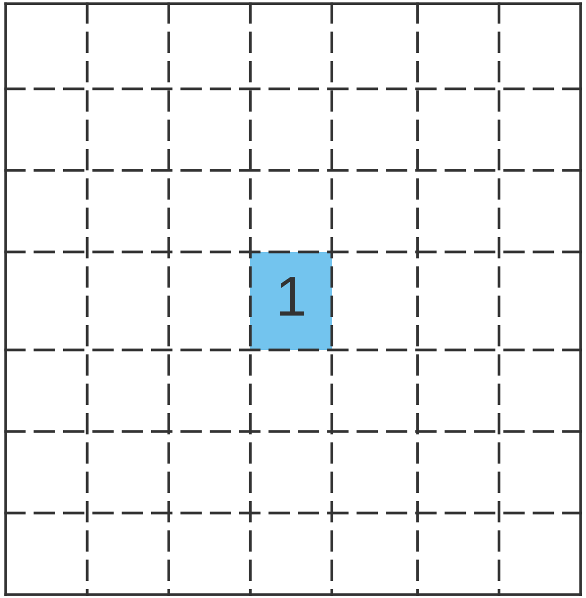
Recommender System

Recommendations

Context

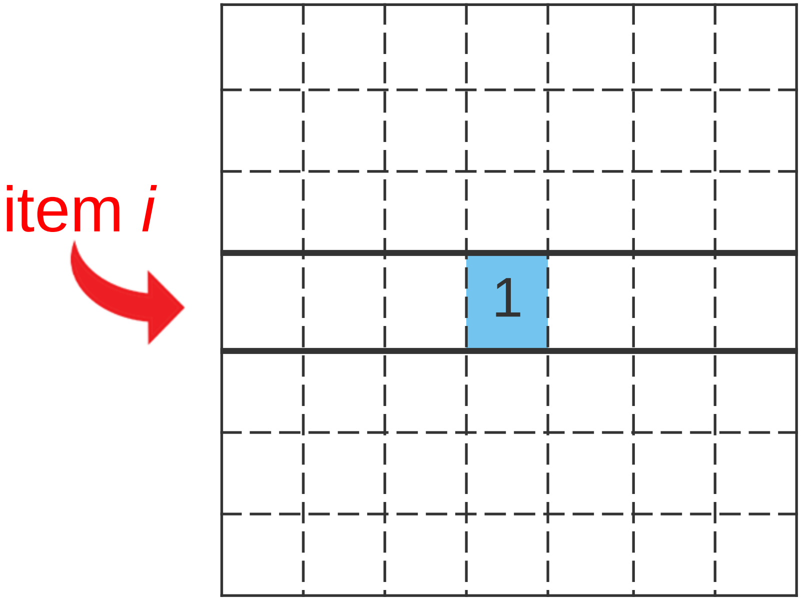
(about users & items)

Item Content Matrix (ICM)

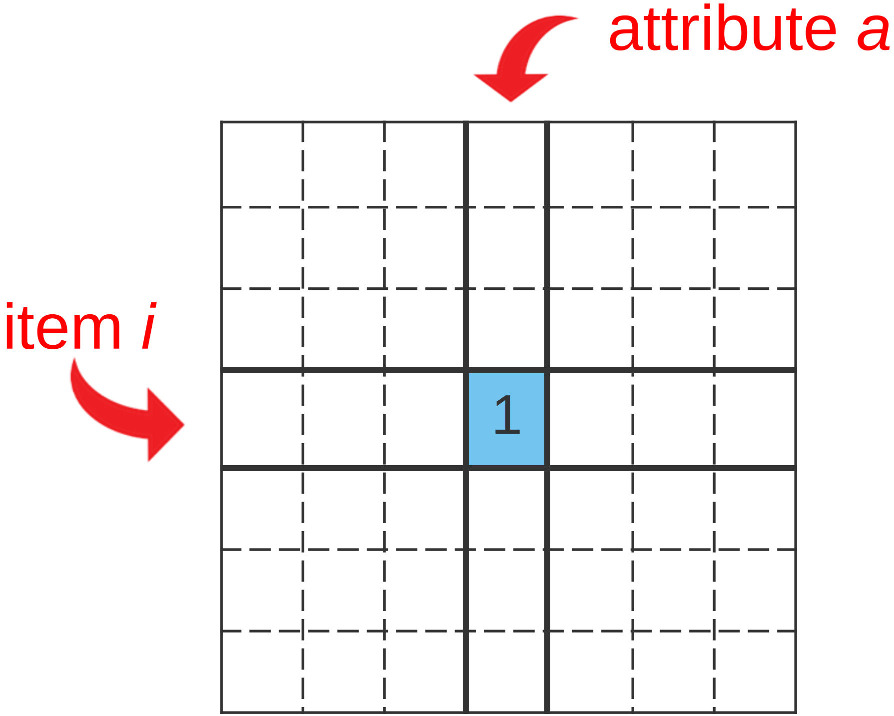


			1			

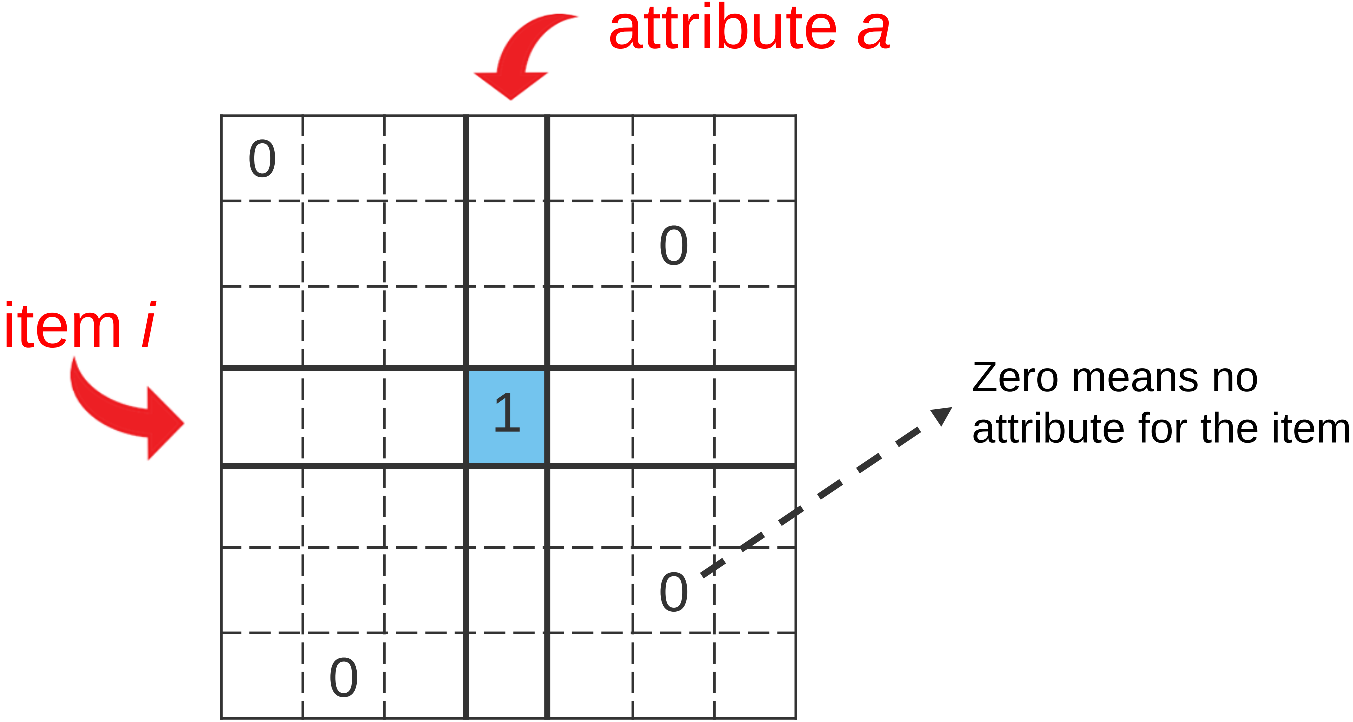
Item Content Matrix (ICM)



Item Content Matrix (ICM)



Item Content Matrix (ICM)

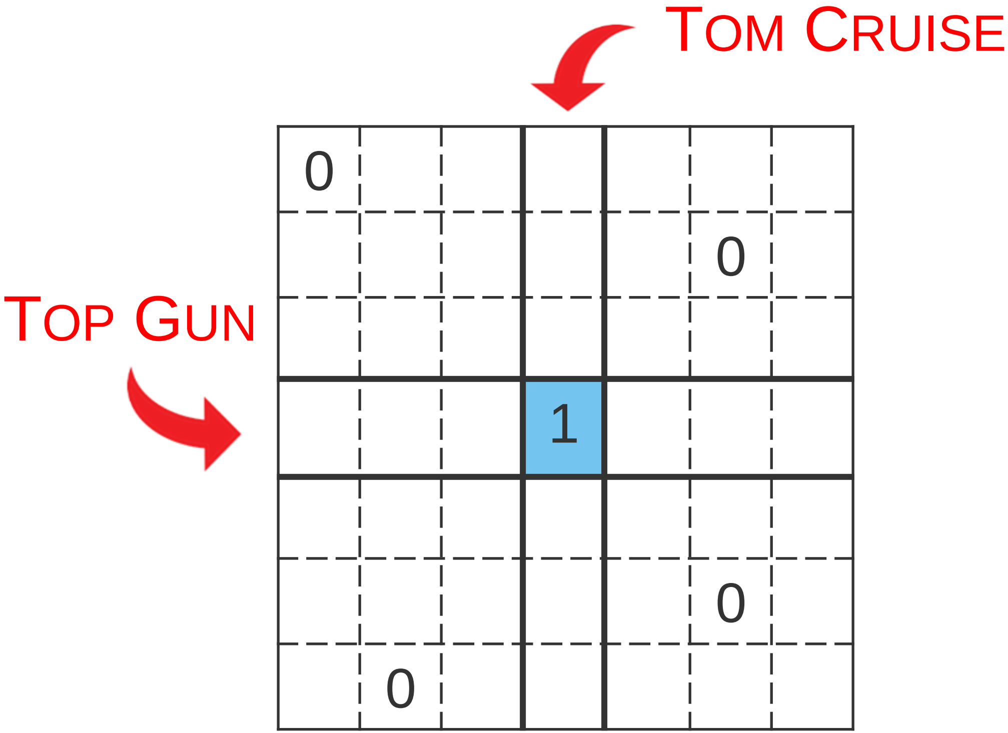


attribute *a*

item *i*

Zero means no attribute for the item

Item Content Matrix (ICM)

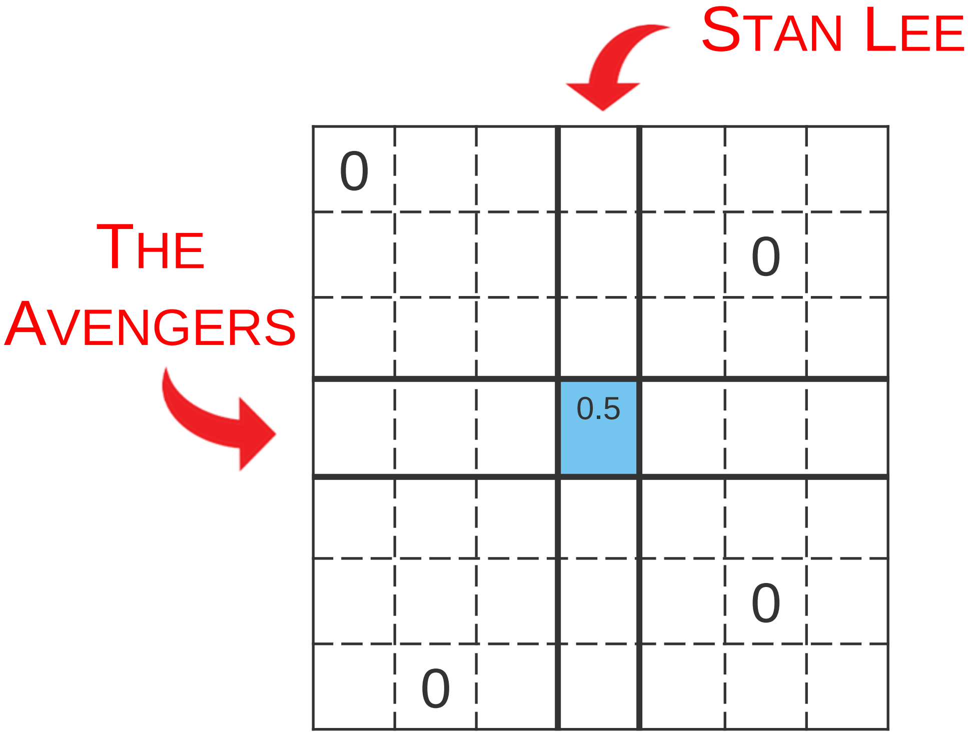


TOM CRUISE

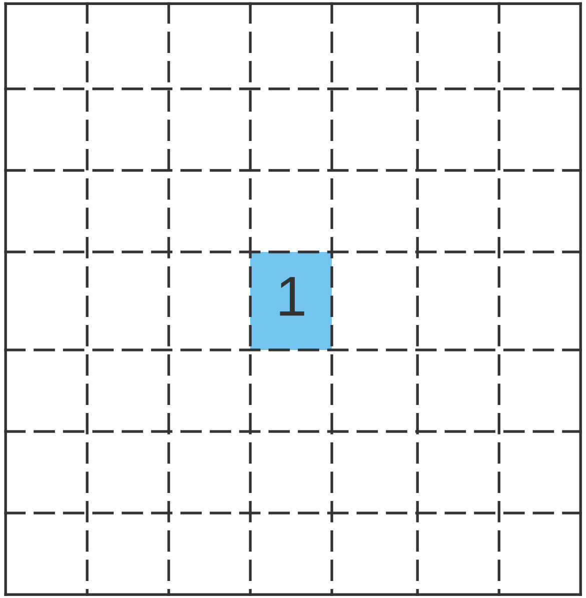
TOP GUN

0						
					0	
			1			
					0	
	0					

Item Content Matrix (ICM)

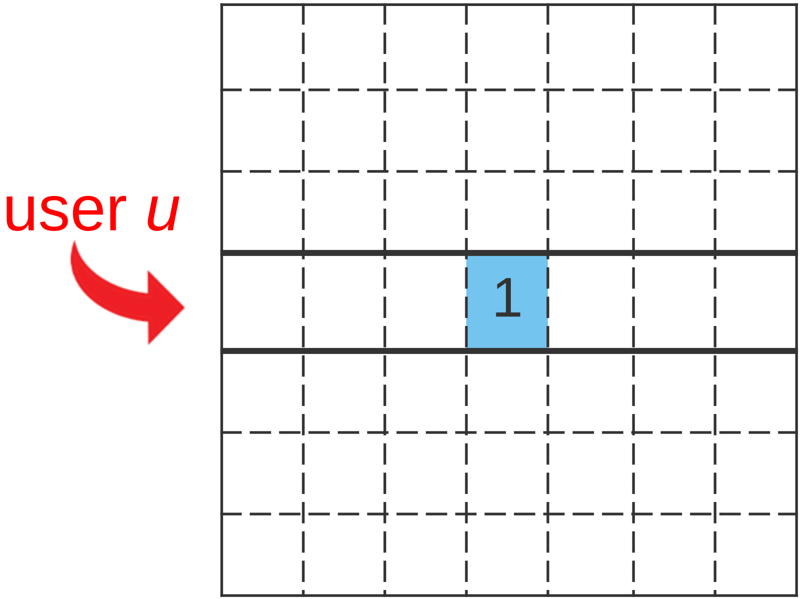


User Rating Matrix (URM)

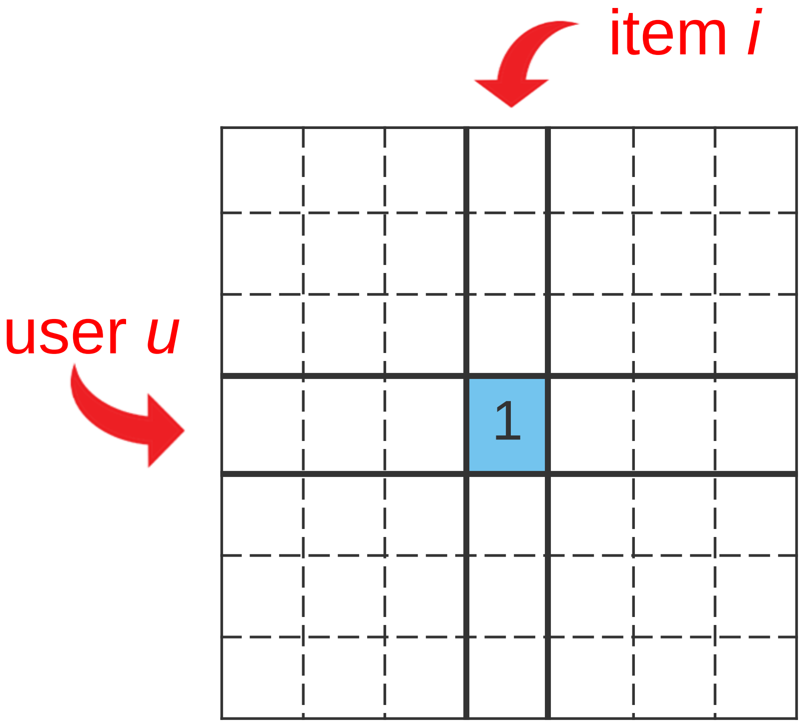


			1			

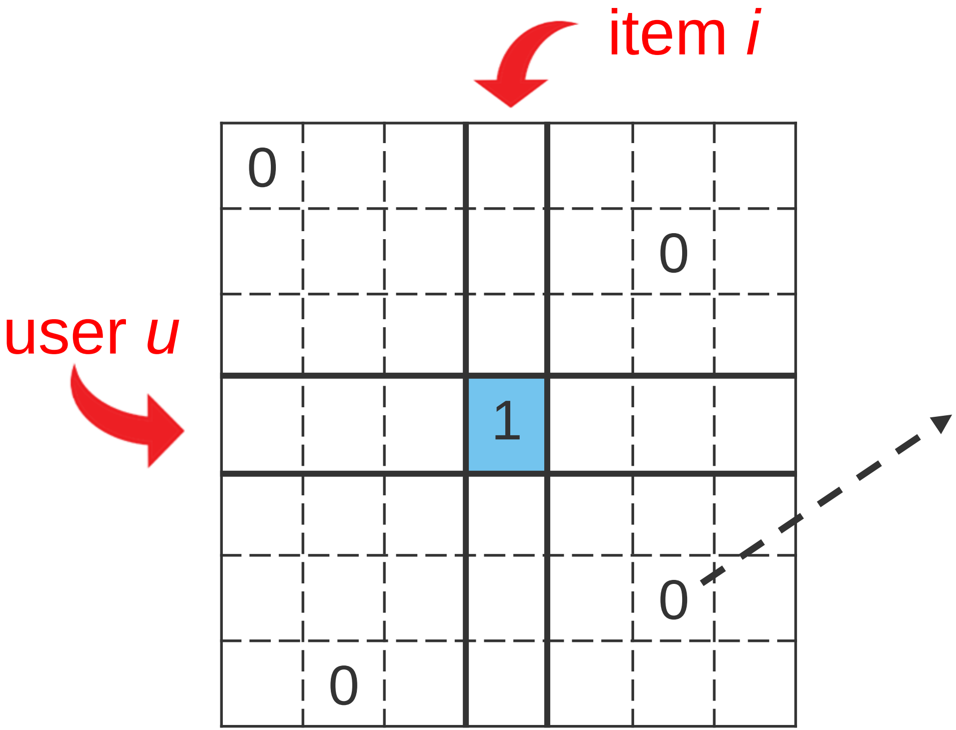
User Rating Matrix (URM)



User Rating Matrix (URM)



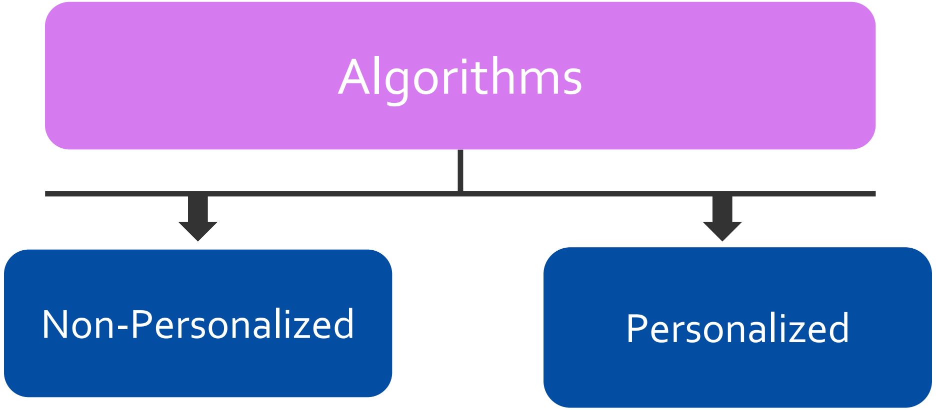
User Rating Matrix (URM)



Zero means no information about the interaction between user and item

Taxonomy of Recommender Systems

Taxonomy of Rec. Systems

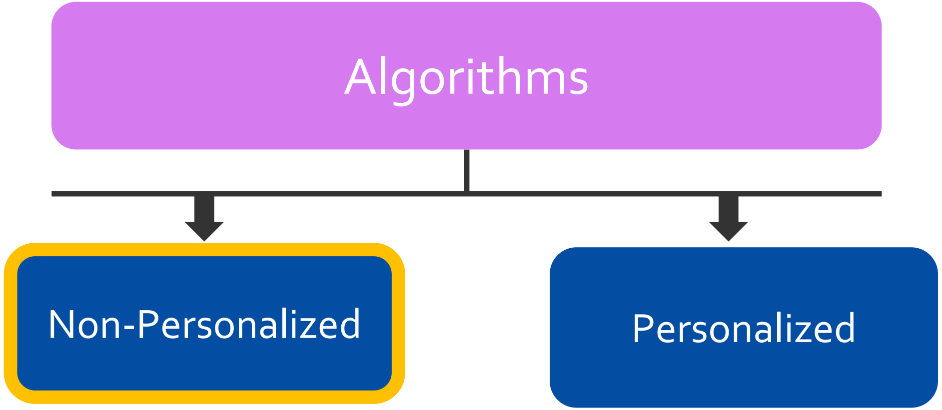


Algorithms

Non-Personalized

Personalized

Taxonomy of Rec. Systems

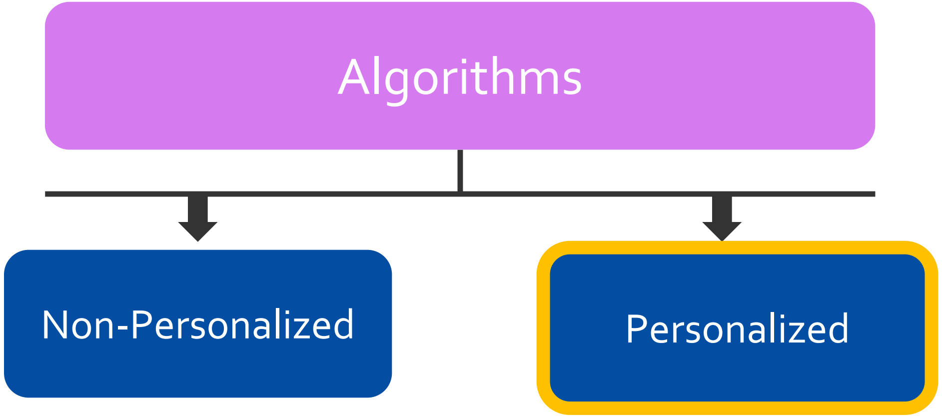


Algorithms

Non-Personalized

Personalized

Taxonomy of Rec. Systems

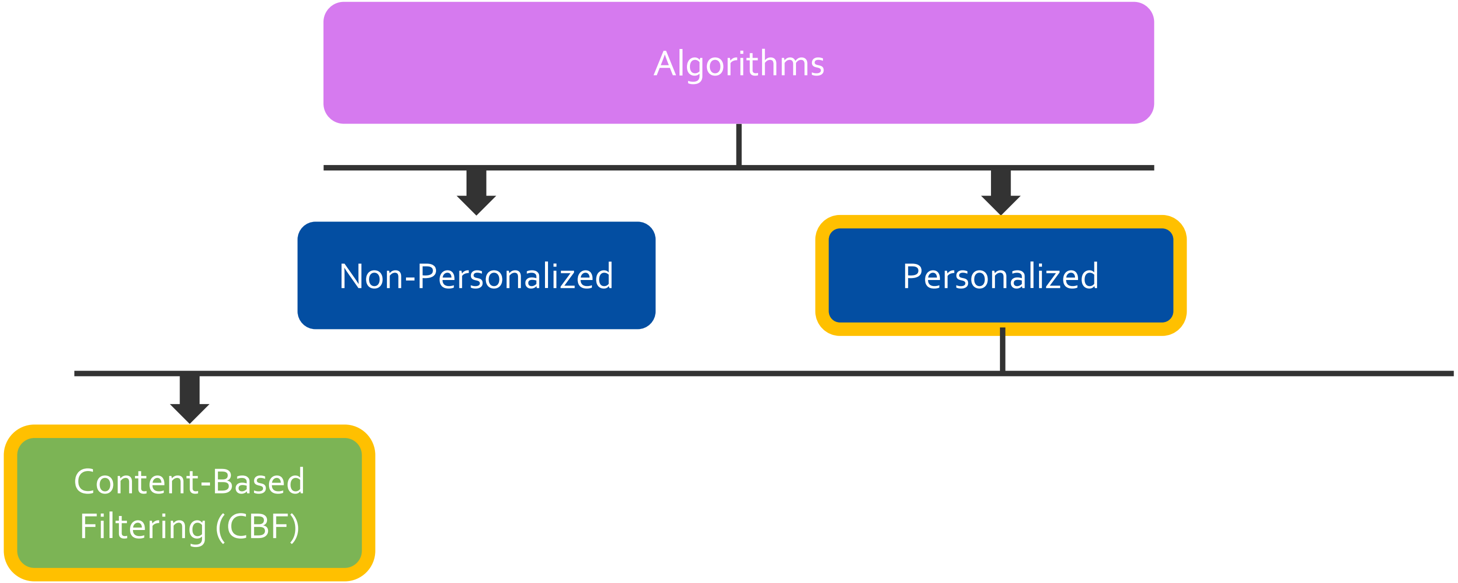


Algorithms

Non-Personalized

Personalized

Taxonomy of Rec. Systems



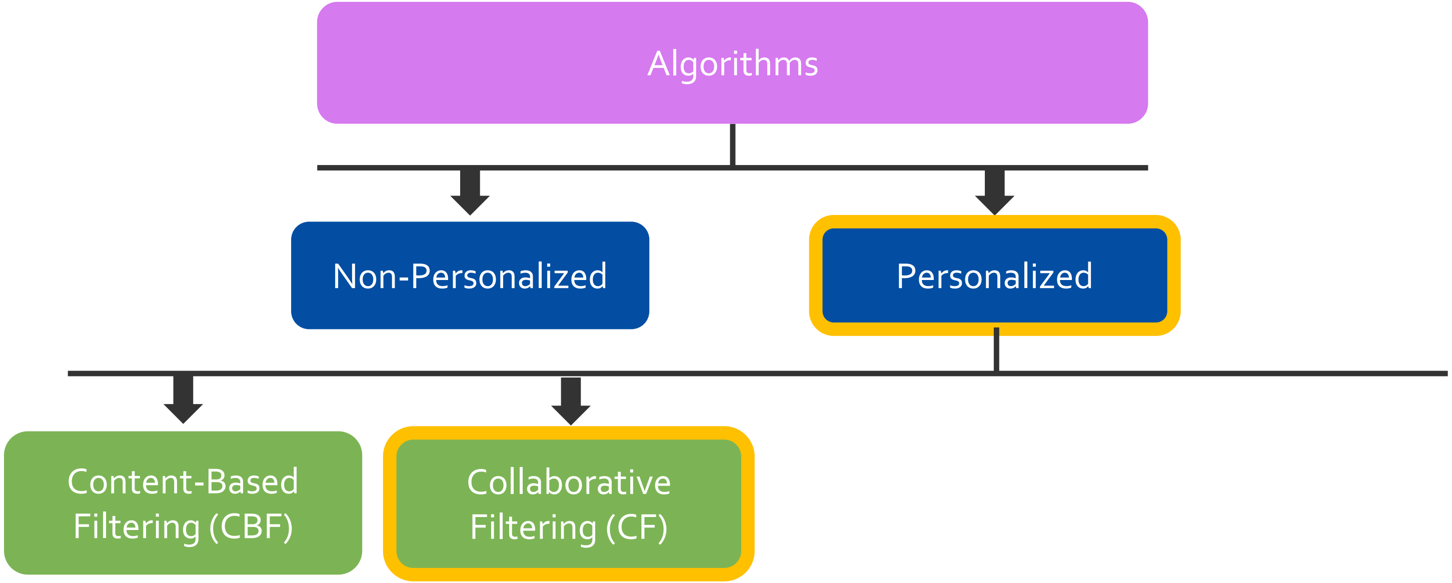
Algorithms

Non-Personalized

Personalized

Content-Based Filtering (CBF)

Taxonomy of Rec. Systems


Algorithms

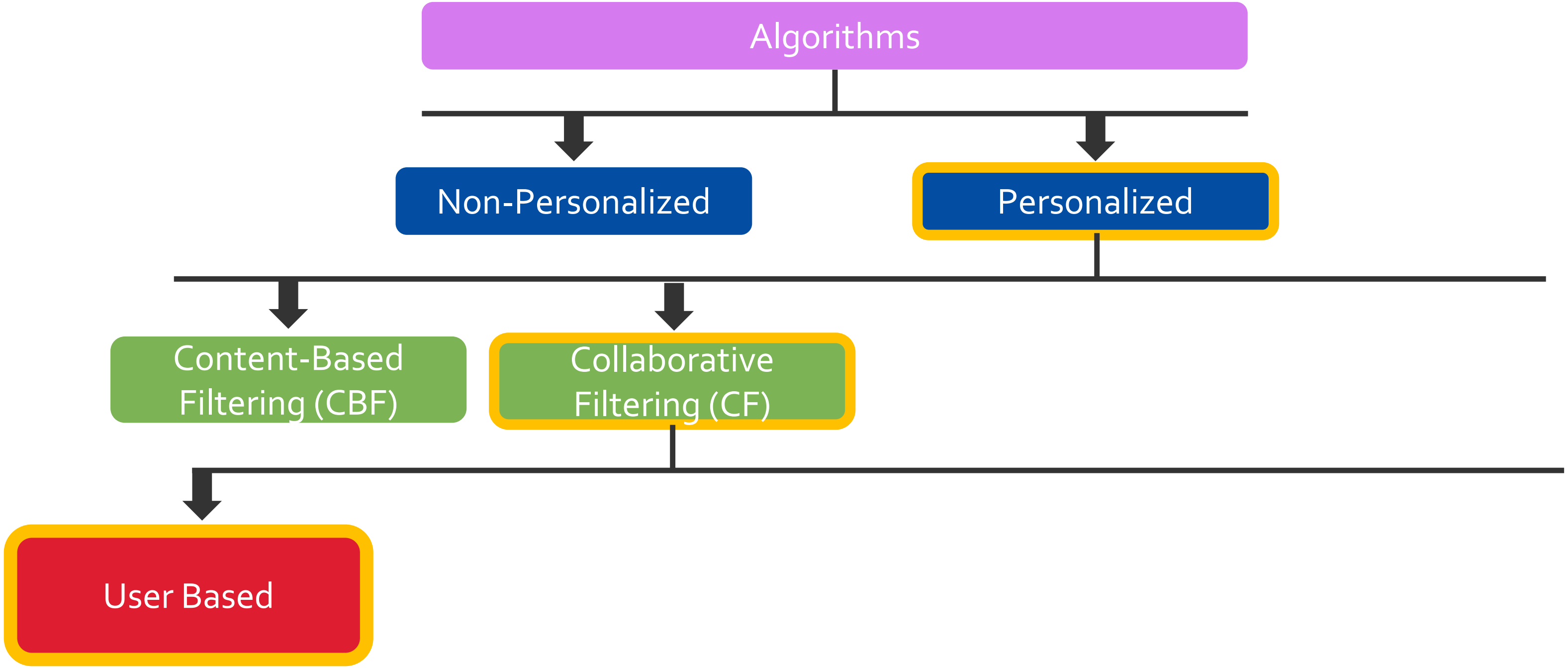
Non-Personalized

Personalized

Content-Based Filtering (CBF)

Collaborative Filtering (CF)

Taxonomy of Rec. Systems



Algorithms

Non-Personalized

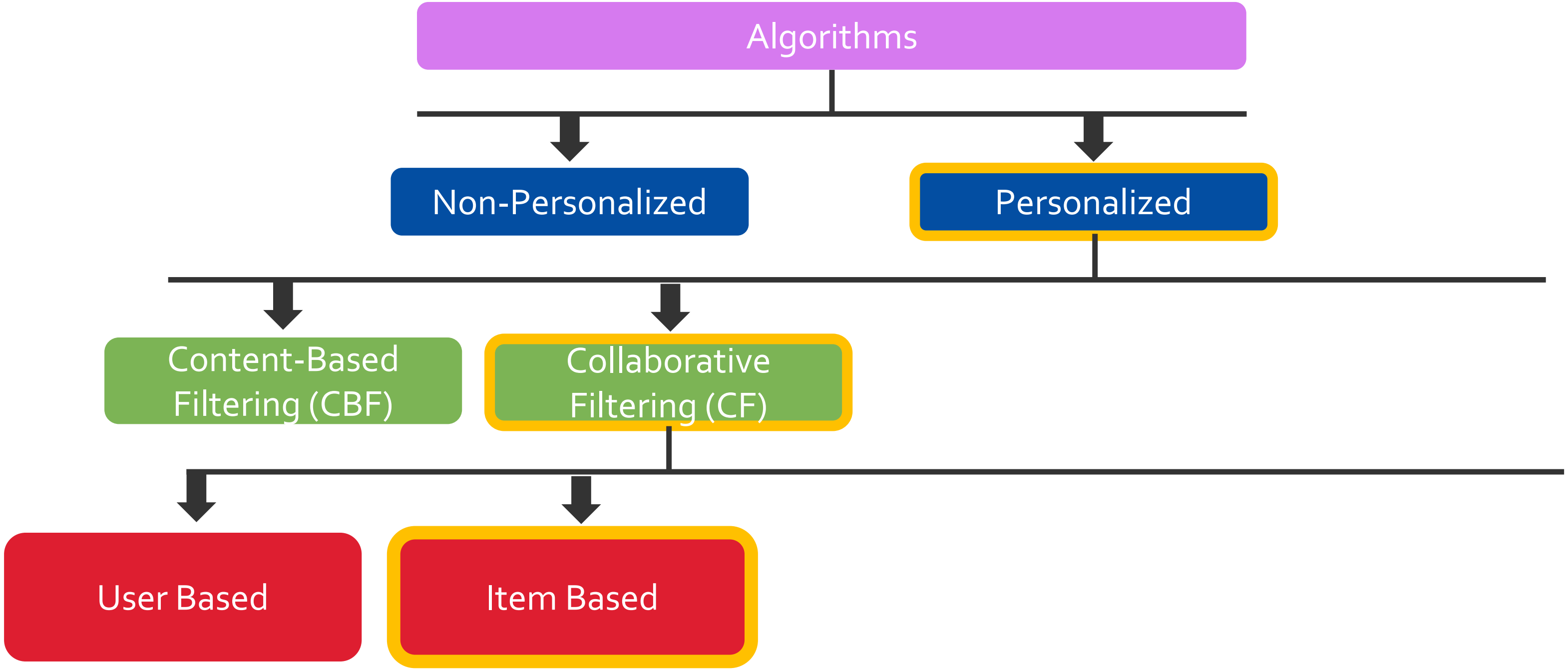
Personalized

Content-Based Filtering (CBF)

Collaborative Filtering (CF)

User Based

Taxonomy of Rec. Systems


Algorithms

Non-Personalized

Personalized

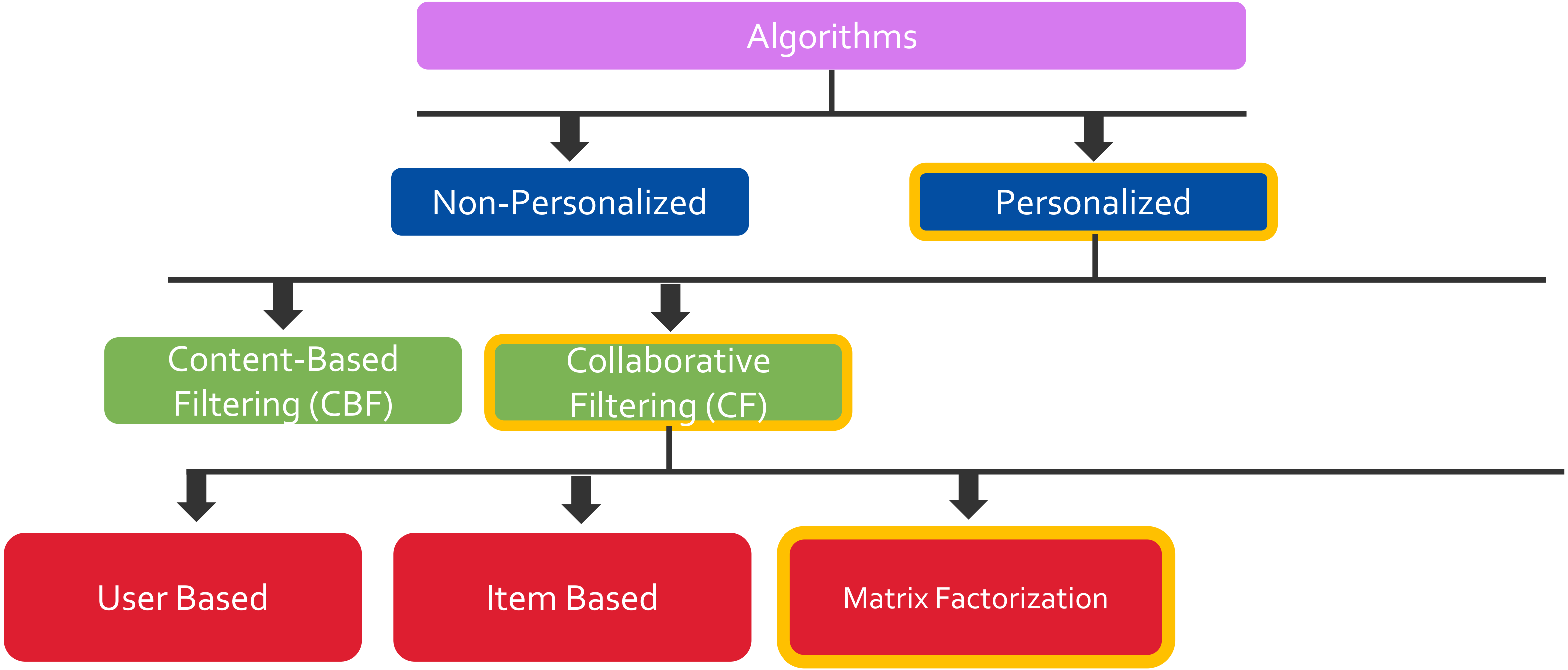
Content-Based Filtering (CBF)

Collaborative Filtering (CF)

User Based

Item Based

Taxonomy of Rec. Systems



Algorithms

Non-Personalized

Personalized

Content-Based Filtering (CBF)

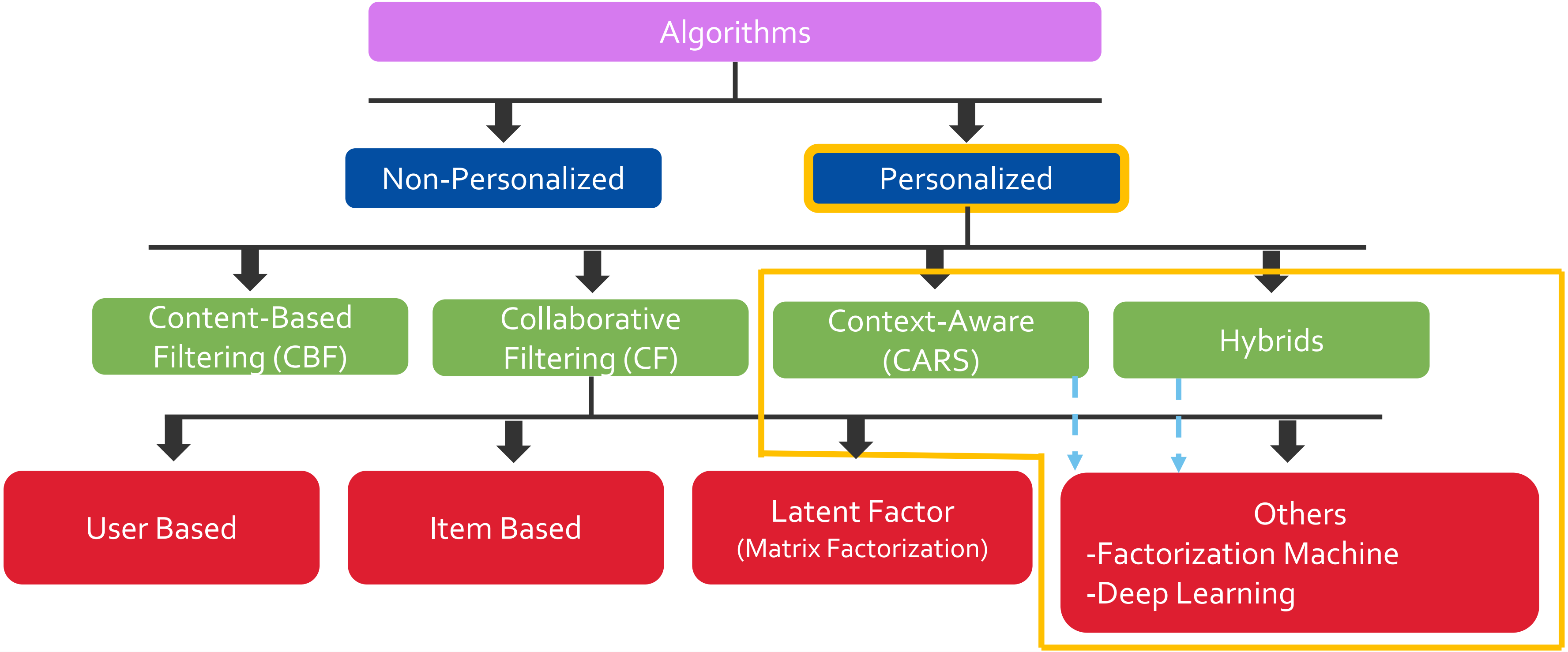
Collaborative Filtering (CF)

User Based

Item Based

Matrix Factorization

Taxonomy of Rec. Systems



Algorithms

Non-Personalized

Personalized

Content-Based Filtering (CBF)

Collaborative Filtering (CF)

Context-Aware (CARS)

Hybrids

User Based

Item Based

Latent Factor
(Matrix Factorization)

Others
-Factorization Machine
-Deep Learning

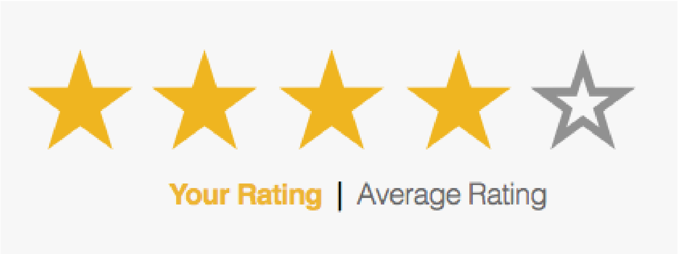
Ratings, predictions and recommendations

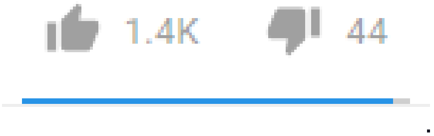
Rating Systems

Explicit ratings

Rating Systems

Explicit ratings





Ratings Distribution

3.0 ●●●○○ 51 reviews

Excellent	11%
Very good	25%
Average	32%
Poor	15%
Terrible	17%

Rating Systems

Explicit ratings

Implicit ratings

Even Ratings Scale

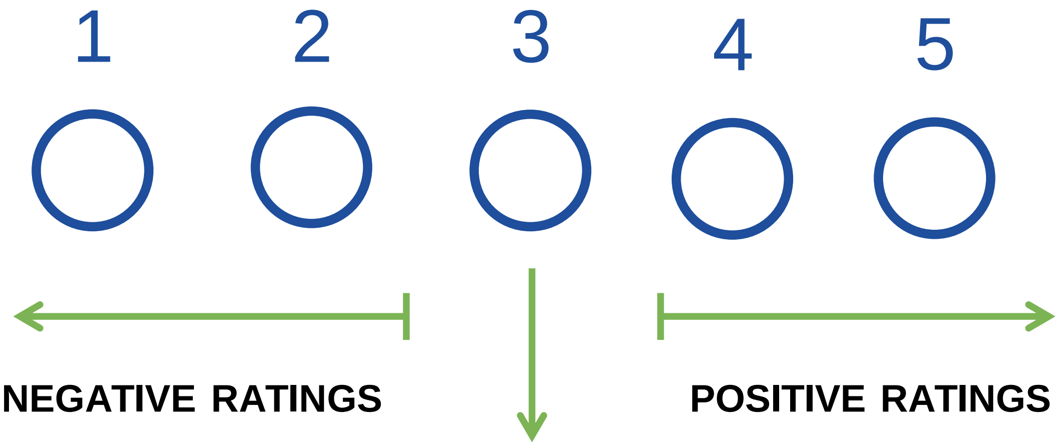
1 2 3 4

○ ○ ○ ○

NEGATIVE RATINGS

POSITIVE RATINGS

Odd Ratings Scale



1 2 3 4 5

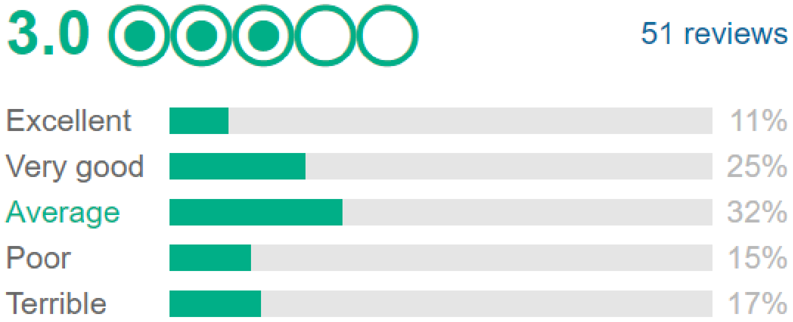
○ ○ ○ ○ ○

NEGATIVE RATINGS

POSITIVE RATINGS

NEUTRAL RATING

Ratings Distribution



3.0 ●●●○○ 51 reviews

Excellent	11%
Very good	25%
Average	32%
Poor	15%
Terrible	17%

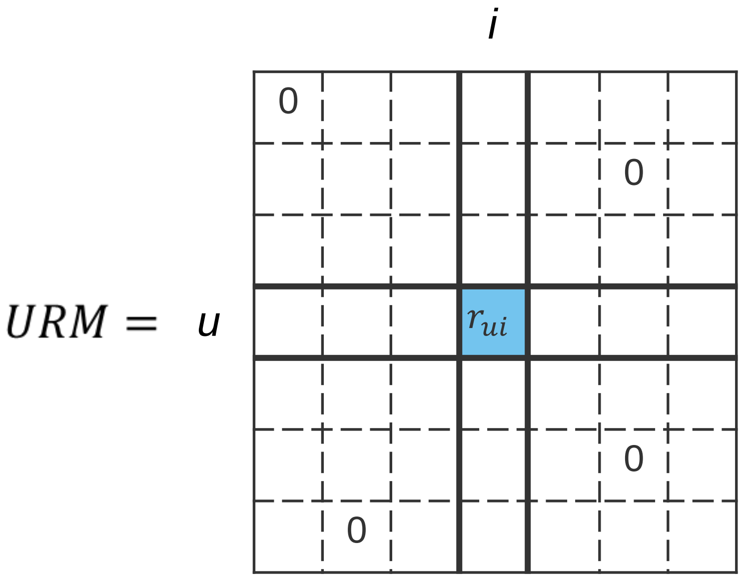
Rating Systems

Explicit ratings

Implicit ratings

- Viewing time of a movie
- Song streaming count
- Purchases
- Other user activity (e.g. opened links)

Inferring Preferences

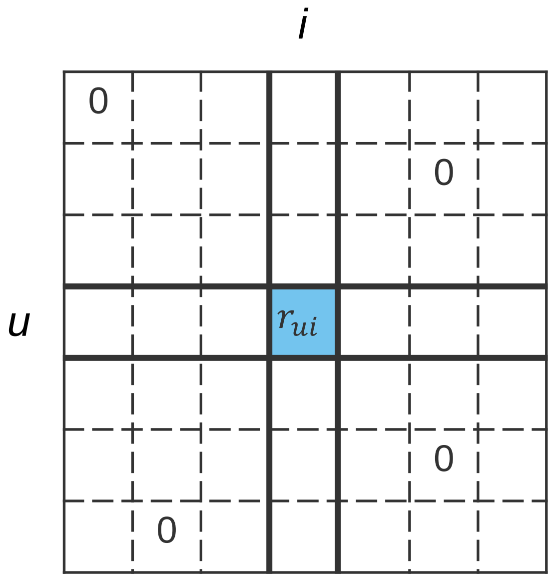


$$URM = \begin{array}{c|ccccccc} & & & & i & & & \\ \hline & 0 & & & & & & \\ & & & & & & 0 & \\ & & & & & & & \\ u & & & & r_{ui} & & & \\ & & & & & & & \\ & & & & & & 0 & \\ & & 0 & & & & & \end{array}$$

r_{ui} = Rating that user *u* gave to item *i*

Inferring Preferences

$URM =$

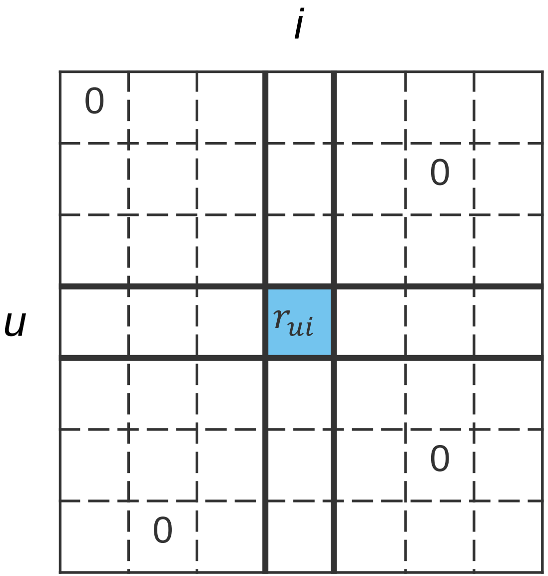


$r_{ui} \in \{0,1\}$ (Implicit)

r_{ui} = Rating that user *u* gave to item *i*

Inferring Preferences

$URM =$



$r_{ui} \in \{0,1\} \leftarrow$ implicit

$r_{ui} \in \{1,2,3,4,5\} \leftarrow$ explicit

r_{ui} = Rating that user *u* gave to item *i*

URM Density

typical URM density $< \mathbf{0.01}\ \%$

URM Density

typical URM density $< 0.01\,\%$

Netflix URM density ≈ 0.002%

MovieLens URM density ≈ 0.005%

Non-Personalized Recommenders

Non-Personalized Recommenders

Top Popular

Non-Personalized Recommenders

Top Popular

$$\begin{array}{c|ccccc} & & & i & & \\ \hline & 3 & 0 & 0 & 1 & 2 \\ & 4 & 2 & 0 & 2 & 3 \\ u & 0 & 0 & 0 & 1 & 2 \\ & 0 & 0 & 5 & 0 & 0 \\ & 3 & 0 & 1 & 0 & 5 \end{array}$$

Non-Personalized Recommenders

Top Popular

i

u

3	0	0	1	2
4	2	0	2	3
0	0	0	1	2
0	0	5	0	0
3	0	1	0	5

3	1	1	3	4

 # ratings per item

Non-Personalized Recommenders

Top Popular

$$\begin{array}{c|ccccc}
 & & & i & & k \\
\hline
 & 3 & 0 & 0 & 1 & 2 \\
 & 4 & 2 & 0 & 2 & 3 \\
u & 0 & 0 & 0 & 1 & 2 \\
 & 0 & 0 & 5 & 0 & 0 \\
 & 3 & 0 & 1 & 0 & 5
\end{array}$$

3	1	1	3	4

item k is the top popular

↓

largest number of ratings

Non-Personalized Recommenders

Top Popular

Best Rated

Non-Personalized Recommenders

Top Popular

Best Rated

$$\begin{array}{c|ccccc} & & & i & & \\ \hline & 3 & 0 & 0 & 1 & 2 \\ & 4 & 2 & 0 & 2 & 3 \\ u & 0 & 0 & 0 & 1 & 2 \\ & 0 & 0 & 5 & 0 & 0 \\ & 3 & 0 & 1 & 0 & 5 \end{array}$$

Non-Personalized Recommenders

Top Popular

Best Rated

$$\begin{array}{c|ccccc} & & & i & & \\ \hline & 3 & 0 & 0 & 1 & 2 \\ & 4 & 2 & 0 & 2 & 3 \\ u & 0 & 0 & 0 & 1 & 2 \\ & 0 & 0 & 5 & 0 & 0 \\ & 3 & 0 & 1 & 0 & 5 \end{array}$$

3.3	2	3	1.3	3	**Avg ratings per item**

Non-Personalized Recommenders

Top Popular

Best Rated

	j		*i*		
	3	0	0	1	2
	4	2	0	2	3
u	0	0	0	1	2
	0	0	5	0	0
	3	0	1	0	5

3.3	2	3	1.3	3

item j is the best rated

↓

largest average rating

Item bias

Average rating of item *i*

$$b_i = \frac{\sum_u r_{ui}}{N_i}$$

Item bias

Average rating of item *i*

$$b_i = \frac{\sum_u r_{ui}}{N_i}$$

r_{ui} : rating given by user *u* to item *i* (non zero ratings)

N_i : number of users who rated item *i*

Item bias: support

Shrinked avg. rating of item *i*

$$b_i = \frac{\Sigma_u \, r_{ui}}{N_i \; + C}$$

Item bias: support

Shrinked avg. rating of item *i*

$$b_i = \frac{\sum_u r_{ui}}{N_i \ + C}$$

r_{ui} : rating given by user *u* to item *i* (non zero ratings)

N_i : number of users who have rated item *i*

C : shrink term (costant value)

Global Effects

Global Effects: Step 1

Avg. ratings for all items and users

$$\mu = \frac{\Sigma_i \Sigma_u r_{ui}^+}{N^+}$$

Global Effects: Step 1

Avg. ratings for all items and users

$$\mu = \frac{\Sigma_i \Sigma_u r_{ui}^{+}}{N^{+}}$$

μ : overall average of ratings, for all users and all items
r_{ui}^{+} : explicit rating given by user *u* to item *i*
N^{+} : total number of non zero ratings

Note: the + symbol denotes that we are not computing this average on the full URM, but ***only*** *on the* ***non zero elements***

Global Effects: Step 2

Normalized rating

$$r'_{ui} = r^{+}_{ui} - \mu$$

to be computed for each user *u* and item *i*, only on non-zero ratings

Global Effects: Step 3

Item bias

$$b_i = \frac{\Sigma_u r'_{ui}}{N_i + C}$$

Global Effects: Step 3

Item bias

$$b_i = \frac{\Sigma_u r'_{ui}}{N_i + C}$$

N_i : number of users who have rated item *i*

to be computed for each item *i*

Global Effects: Step 4

Recompute rating

$$r''_{ui} = r'_{ui} - b_i$$

to be computed for each user *u* and item *i*, only on non-zero ratings

Global Effects: Step 5

User bias

$$b_u = \frac{\Sigma_i \, r''_{ui}}{N_u + C}$$

Global Effects: Step 5

User bias

$$b_u = \frac{\sum_i r''_{ui}}{N_u + C}$$

N_u : number of items *i* rated by user *u*

to be computed for each user *u*

Global Effects: Step 6

Global effects final formula estimated rating

$$\tilde{r}_{ui} = \mu + b_i + b_u$$

to be computed for each user *u* and item *i*, only on non-zero ratings

Global Effects: Recap

- Step 1: compute the average of all ratings (μ)

$$\tilde{r}_{ui} = \mu + b_i + b_u$$

Global Effects: Recap

- Step 1: compute the average of all ratings (μ)
- Step 2: remove this quantity from the URM

$$\tilde{r}_{ui} = \mu + b_i + b_u$$

Global Effects: Recap

- Step 1: compute the average of all ratings (μ)
- Step 2: remove this quantity from the URM
- Step 3: compute the bias for each item (b_i)

$$\tilde{r}_{ui} = \mu + b_i + b_u$$

Global Effects: Recap

- Step 1: compute the average of all ratings (μ)
- Step 2: remove this quantity from the URM
- Step 3: compute the bias for each item (b_i)
- Step 4: remove this quantity from the URM

$$\tilde{r}_{ui} = \mu + b_i + b_u$$

Global Effects: Recap

- Step 1: compute the average of all ratings (μ)
- Step 2: remove this quantity from the URM
- Step 3: compute the bias for each item (b_i)
- Step 4: remove this quantity from the URM
- Step 5: compute the bias for each user (b_u)

$$\tilde{r}_{ui} = \mu + b_i + b_u$$

Global Effects: Recap

- Step 1: compute the average of all ratings (μ)
- Step 2: remove this quantity from the URM
- Step 3: compute the bias for each item (b_i)
- Step 4: remove this quantity from the URM
- Step 5: compute the bias for each user (b_u)
- Step 6: final formula creating a new URM

$$\tilde{r}_{ui} = \mu + b_i + b_u$$

Evaluation of Recommender Systems

Requirements

FUNCTIONAL REQUIREMENTS

Requirements

FUNCTIONAL REQUIREMENTS

What the software does

Requirements

FUNCTIONAL REQUIREMENTS

What the software does

NON-FUNCTIONAL REQUIREMENTS

Requirements

FUNCTIONAL REQUIREMENTS

What the software does

NON-FUNCTIONAL REQUIREMENTS

How the software does its job

Non-Functional Requirements

RESPONSE TIME

Non-Functional Requirements

RESPONSE TIME

- How long does it take for the system to generate one recommendation?

Non-Functional Requirements

RESPONSE TIME

SCALABILITY

Non-Functional Requirements

RESPONSE TIME

SCALABILITY

- How many recommendations per second the system is able to generate?

Non-Functional Requirements

RESPONSE TIME

SCALABILITY

PRIVACY AND SECURITY

Non-Functional Requirements

RESPONSE TIME

SCALABILITY

PRIVACY AND SECURITY

- Protect against reverse engineering
- Protect against intrusions from outside

Non-Functional Requirements

RESPONSE TIME

SCALABILITY

PRIVACY AND SECURITY

USER INTERFACE

Non-Functional Requirements

RESPONSE TIME

SCALABILITY

PRIVACY AND SECURITY

USER INTERFACE

- Which is the best place to show recommendations?
- How many items should be recommended?

Quality indicators for Recommender Systems

Quality Indicators

RELEVANCE

Quality Indicators

RELEVANCE

- Recommend items that users like

Quality Indicators

RELEVANCE

COVERAGE

Quality Indicators

RELEVANCE

COVERAGE

- Ability to recommend most of the items in a catalogue

Quality Indicators

RELEVANCE

COVERAGE

NOVELTY

Quality Indicators

RELEVANCE

COVERAGE

NOVELTY

- Recommend items unkown to the user

Quality Indicators

RELEVANCE

COVERAGE

NOVELTY

DIVERSITY

Quality Indicators

RELEVANCE

COVERAGE

NOVELTY

DIVERSITY

- Diversify the items recommended

Quality Indicators

RELEVANCE

COVERAGE

NOVELTY

DIVERSITY

CONSISTENCY

Quality Indicators

RELEVANCE

COVERAGE

NOVELTY

DIVERSITY

CONSISTENCY

- Recommendations should not change to often

Quality Indicators

RELEVANCE

COVERAGE

NOVELTY

DIVERSITY

CONSISTENCY

CONFIDENCE

Quality Indicators

RELEVANCE

COVERAGE

NOVELTY

DIVERSITY

CONSISTENCY

CONFIDENCE

- How much a system is sure about a recommendation

Quality Indicators

RELEVANCE

COVERAGE

NOVELTY

DIVERSITY

CONSISTENCY

CONFIDENCE

SERENDIPITY

Quality Indicators

RELEVANCE

COVERAGE

NOVELTY

DIVERSITY

CONSISTENCY

CONFIDENCE

SERENDIPITY

- The ability of *surprising* the user
- The ability to recommend items that users would have never been able to discover by themselves

Evaluation Techniques

Evaluation Techniques

ONLINE

Evaluation Techniques

ONLINE

OFF-LINE

Online Evaluation

DIRECT USER FEEDBACK

Online Evaluation

DIRECT USER FEEDBACK

Online Evaluation

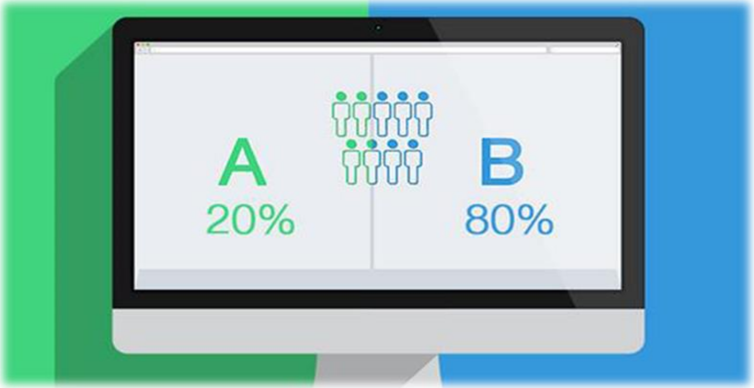
DIRECT USER FEEDBACK

A/B TESTING

Online Evaluation

DIRECT USER FEEDBACK

A/B TESTING



Online Evaluation

DIRECT USER FEEDBACK

A/B TESTING

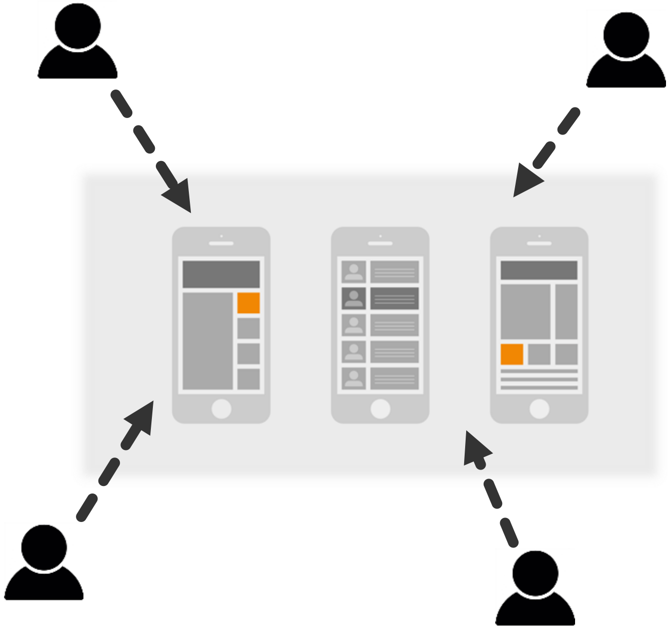
CONTROLLED EXPERIMENTS

Online Evaluation

DIRECT USER FEEDBACK

A/B TESTING

CONTROLLED EXPERIMENTS



Online Evaluation

DIRECT USER FEEDBACK

A/B TESTING

CONTROLLED EXPERIMENTS

CROWDSOURCING

Online Evaluation

DIRECT USER FEEDBACK

A/B TESTING

CONTROLLED EXPERIMENTS

CROWDSOURCING

Off-line Evaluation

TASK

Off-line Evaluation

TASK

DATASET

Off-line Evaluation

TASK

DATASET

PARTITIONING

Off-line Evaluation

TASK

DATASET

PARTITIONING

METRICS

Off-line Evaluation: Task

RATING PREDICTION

Off-line Evaluation: Task

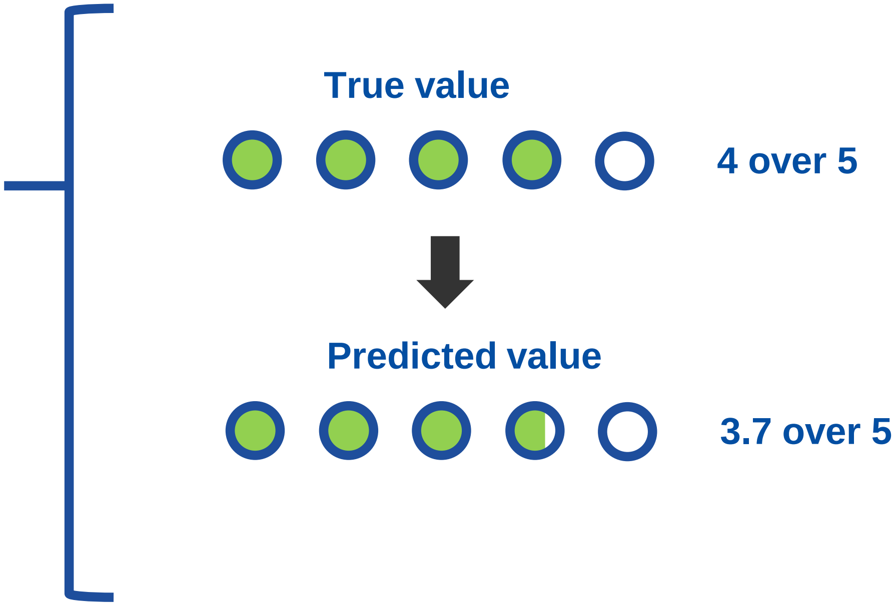
RATING PREDICTION

True value

● ● ● ● ○ **4 over 5**

Off-line Evaluation: Task

RATING PREDICTION



True value

●●●●○ **4 over 5**

Predicted value

●●●◐○ **3.7 over 5**

Off-line Evaluation: Task

RATING PREDICTION

TOP-N RECOMMENDATION

Off-line Evaluation: Task

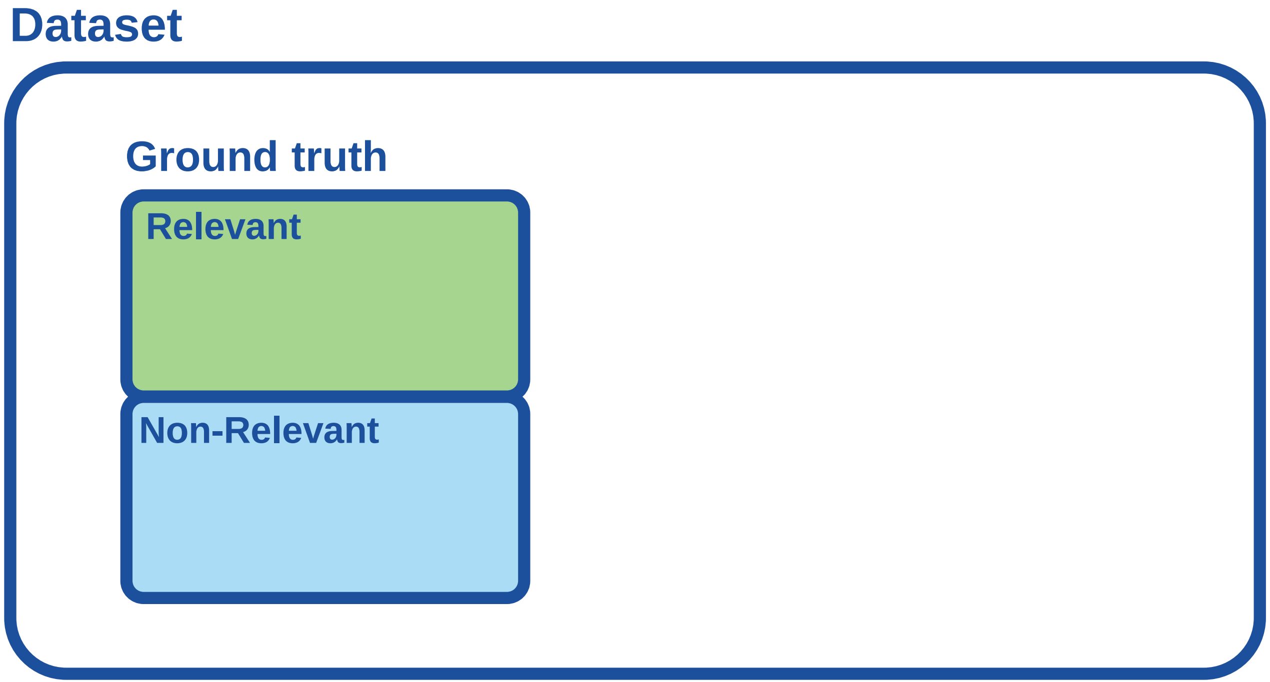
RATING PREDICTION

TOP-N RECOMMENDATION

Off-line Evaluation: Dataset

Dataset

Off-line Evaluation: Dataset



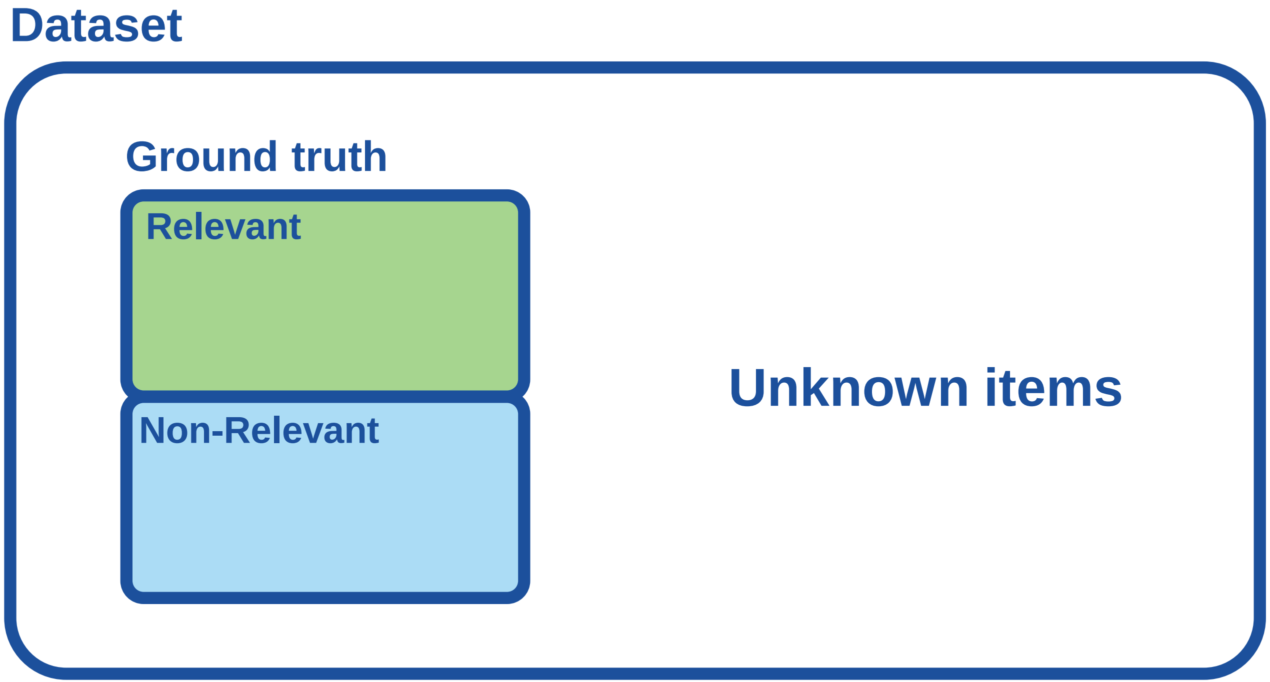
Dataset

Ground truth

Relevant

Non-Relevant

Off-line Evaluation: Dataset



Dataset

Ground truth

Relevant

Non-Relevant

Unknown items

Off-line Evaluation: Partitioning

- Model = f(URM)
- Estimated ratings = g(model, user profile)

Off-line Evaluation: Partitioning

- Model = f(URM)
- Estimated ratings = g(model, user profile)

E.G. {
Model = Star Wars is similar to Avatar
User profile = Paolo Cremonesi likes Star Wars

Off-line Evaluation: Partitioning

- Model = f(URM)
- Estimated ratings = g(model, user profile)

E.G.

Model = Star Wars is similar to Avatar
User profile = Paolo Cremonesi likes Star Wars

- Estimated ratings ↔ True recommendation

Partitioning: Hold Out of Ratings

- Model = f(X)
- Estimated ratings = g(model, Y)
- Estimated ratings ↔ Z

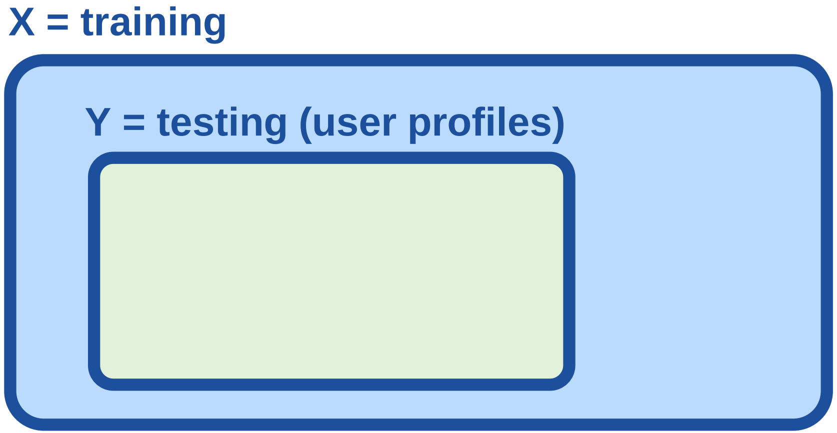
Partitioning: Hold Out

- Model = f(X)
- Estimated ratings = g(model, Y)
- Estimated ratings ↔ Z

X = training

Partitioning: Hold Out of Ratings

- Model = f(X)
- Estimated ratings = g(model, Y)
- Estimated ratings ↔ Z

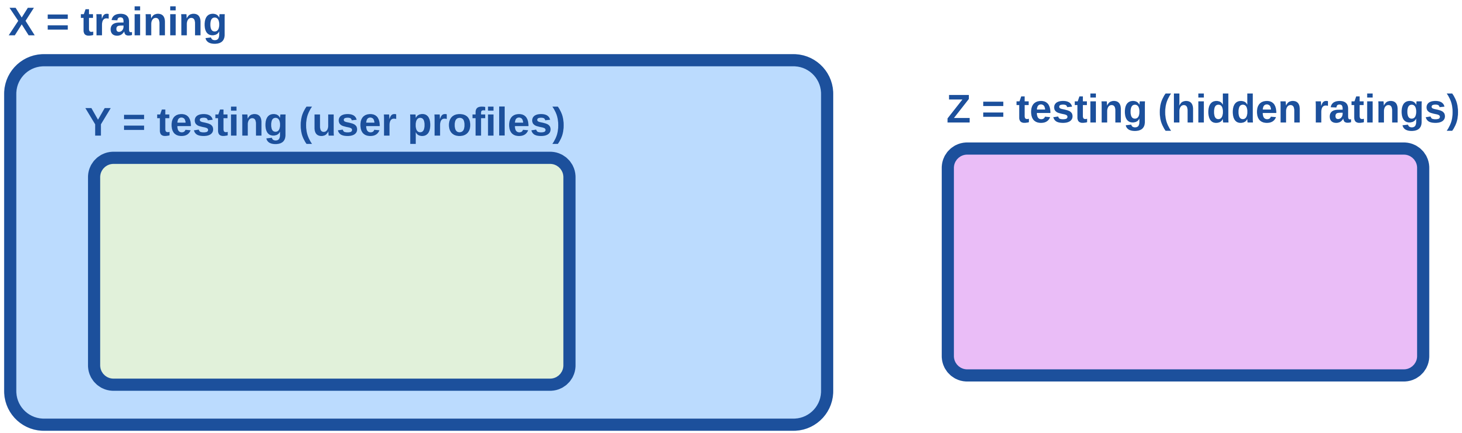


X = training

Y = testing (user profiles)

Partitioning: Hold Out of Ratings

- Model = f(X)
- Estimated ratings = g(model, Y)
- Estimated ratings ↔ Z



X = training

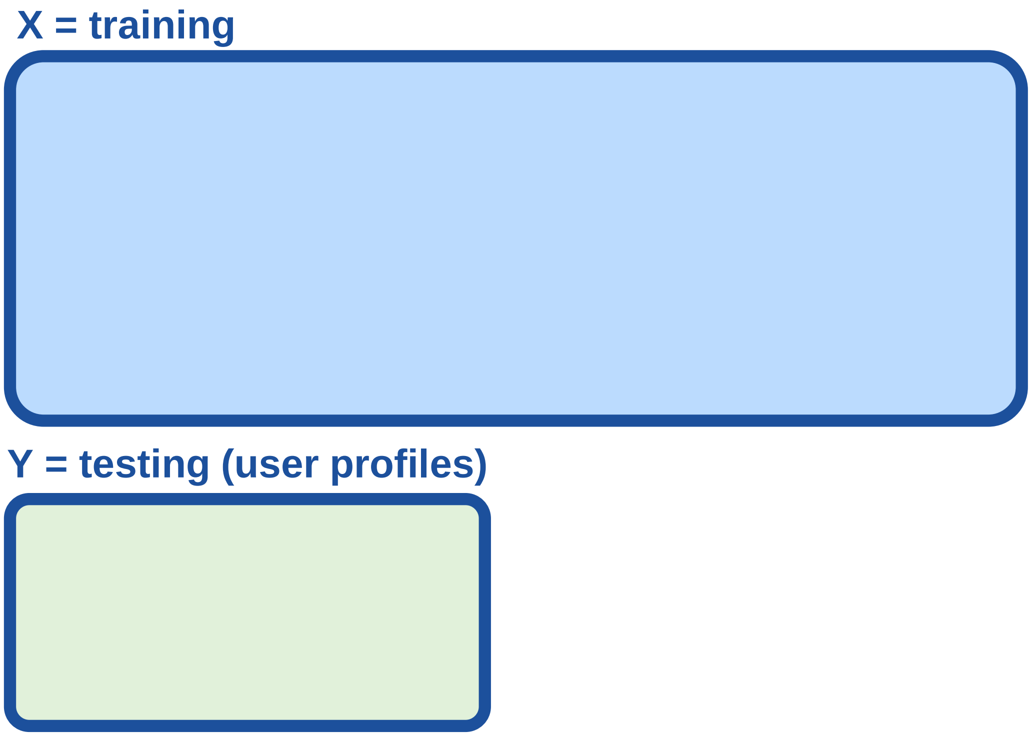
Y = testing (user profiles)

Z = testing (hidden ratings)

Partitioning: Hold Out of Users

X = training

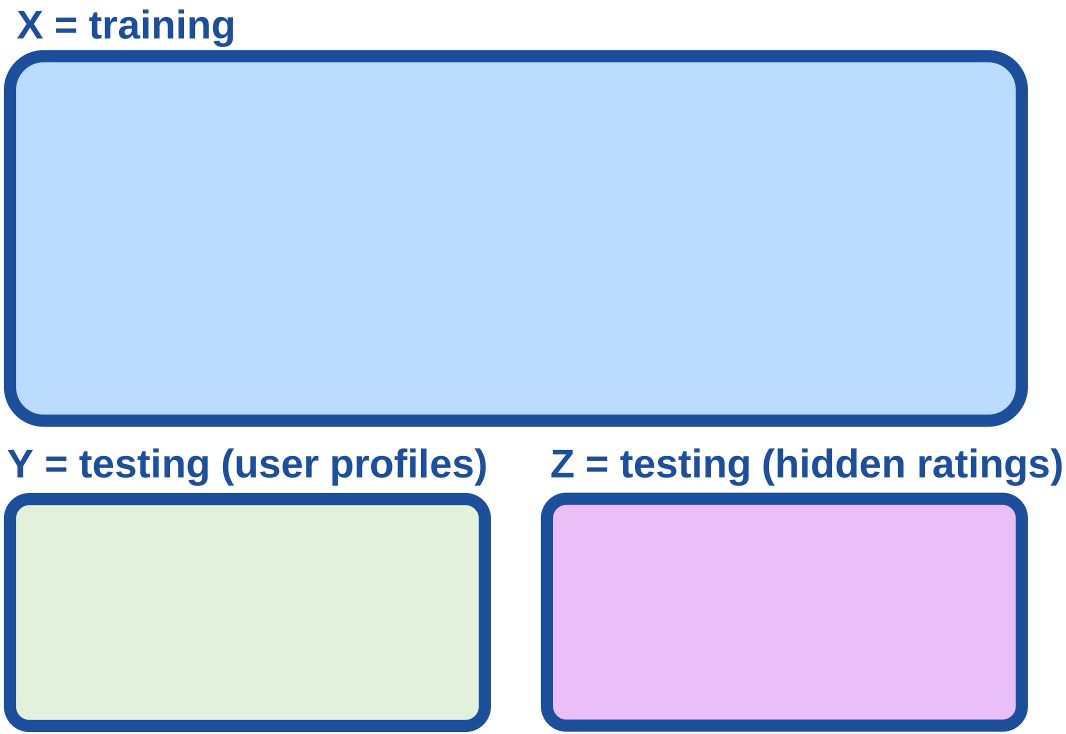
Partitioning: Hold Out of Users



X = training

Y = testing (user profiles)

Partitioning: Hold Out of Users



X = training

Y = testing (user profiles)

Z = testing (hidden ratings)

Off-line Evaluation: Error Metrics

True value r_{ui}

● ● ● ● ○ **4 over 5**

Off-line Evaluation: Error Metrics

True value r_{ui}

● ● ● ● ○ **4 over 5**

Estimated value $\hat{r}_{ui}$

● ● ● ◐ ○ **3.7 over 5**

Off-line Evaluation: Error Metrics

True value r_{ui}

● ● ● ● ○ **4 over 5**

Estimated value $\hat{r}_{ui}$

● ● ● ◐ ○ **3.7 over 5**

Error: $e_{ui} = r_{ui} - \hat{r}_{ui}$

Off-line Evaluation: Error Metrics

True value r_{ui}

● ● ● ● ○ **4 over 5**

Estimated value $\hat{r}_{ui}$

● ● ● ◐ ○ **3.7 over 5**

Error: $e_{ui} = r_{ui} - \hat{r}_{ui} = 4 - 3.7 = 0.3$

Off-line Evaluation: Error Metrics

True value r_{ui}		**Estimated value $\hat{r}_{ui}$**	
● ● ● ● ○	**4 over 5**	● ● ● ◐ ○	**3.7 over 5**

$$\text{Error: } e_{ui} = r_{ui} - \hat{r}_{ui} = 4 - 3.7 = 0.3$$

$\hat{r}_{ui}$: rating estimated by the recommender system
r_{ui}: true rating in the test set

Off-line Evaluation: Error Metrics

Mean absolute error:

$$MAE = \frac{\sum_{u,i \in T} |r_{ui} - \hat{r}_{ui}|}{|T|}$$

Off-line Evaluation: Error Metrics

Mean absolute error:

$$MAE = \frac{\sum_{u,i \in T} |r_{ui} - \hat{r}_{ui}|}{|T|}$$

Mean square error:

$$MSE = \frac{\sum_{u,i \in T} (r_{ui} - \hat{r}_{ui})^2}{|T|}$$

Off-line Evaluation: Error Metrics

Mean absolute error:

$$MAE = \frac{\sum_{u,i \in T} |r_{ui} - \hat{r}_{ui}|}{|T|}$$

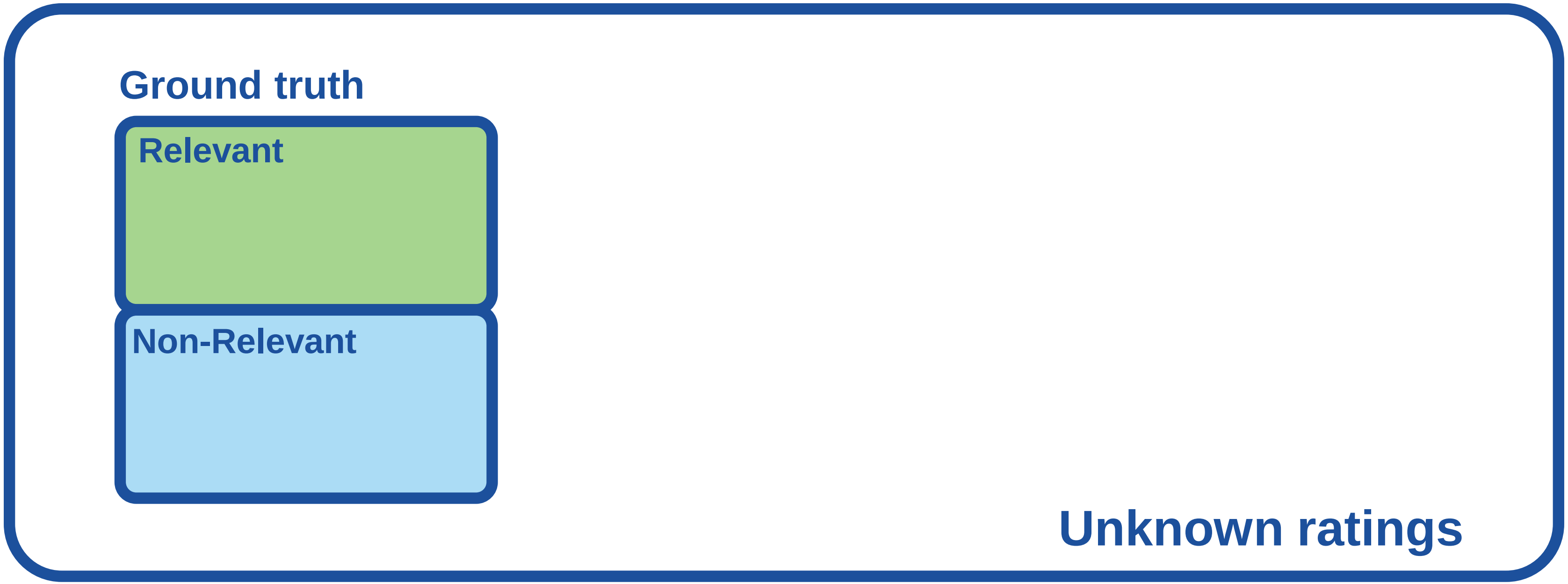
Mean square error:

$$MSE = \frac{\sum_{u,i \in T} (r_{ui} - \hat{r}_{ui})^2}{|T|}$$

T: test set
$\hat{r}_{ui}$: rating estimated by recommender system
r_{ui}: true rating in the test set

Off-line Evaluation: Error Metrics

Dataset



Off-line Evaluation: Error Metrics

Dataset

Ground truth

Recommendations

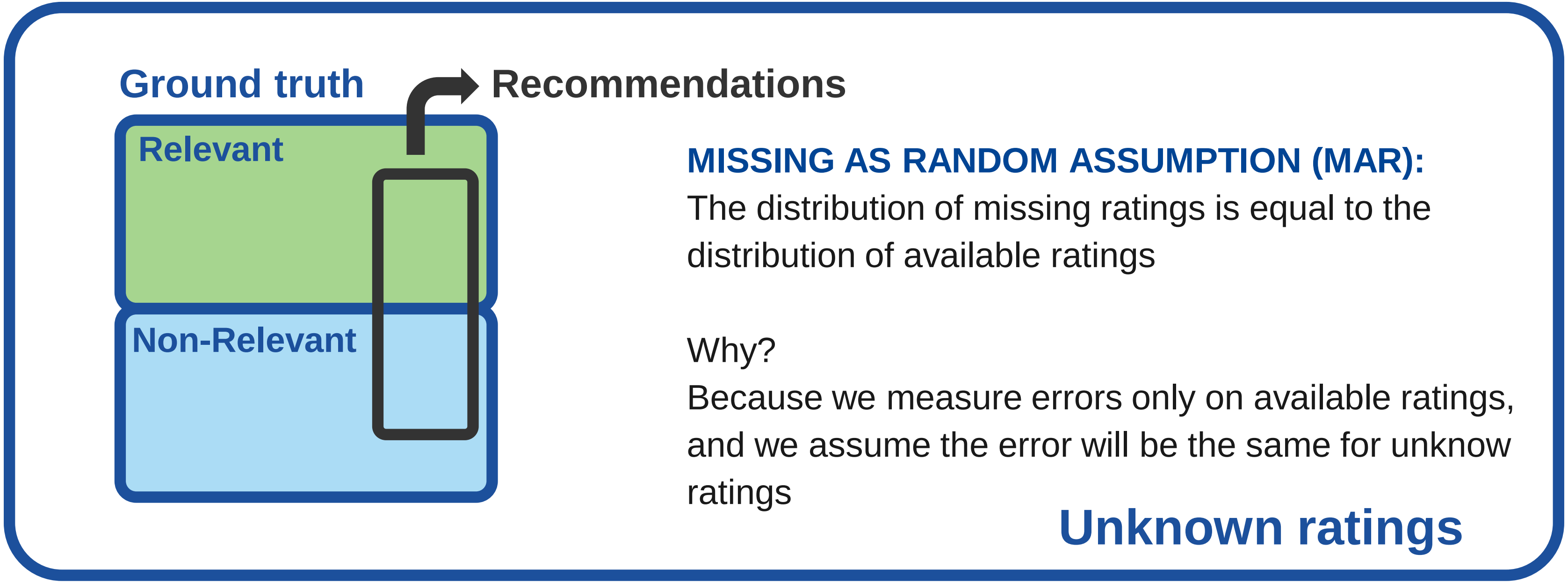
Relevant

Non-Relevant

Unknown ratings

Off-line Evaluation: Error Metrics

Dataset



Ground truth

Recommendations

Relevant

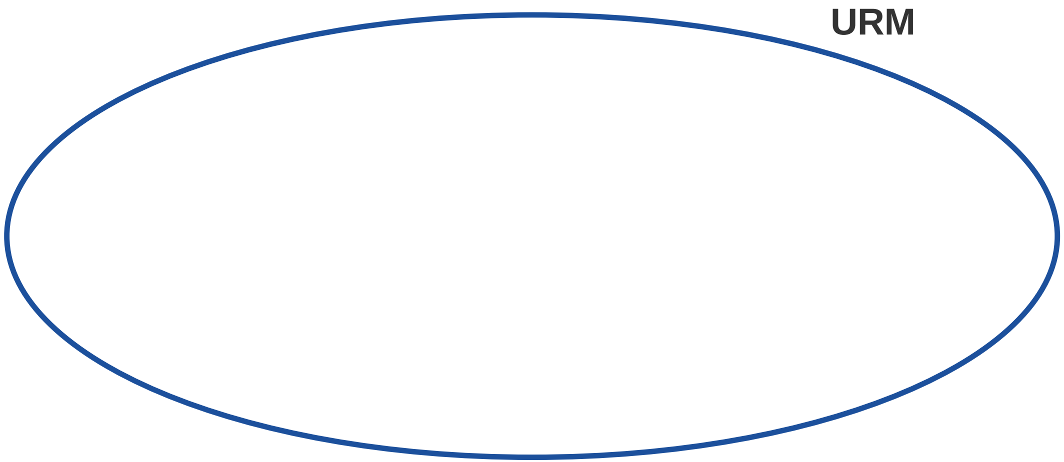
Non-Relevant

MISSING AS RANDOM ASSUMPTION (MAR):
The distribution of missing ratings is equal to the distribution of available ratings

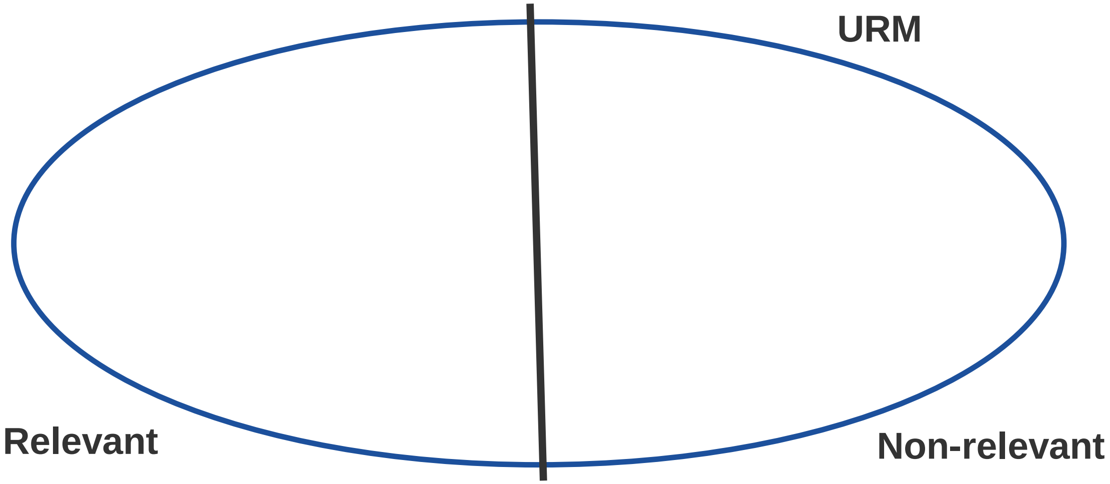
Why?
Because we measure errors only on available ratings, and we assume the error will be the same for unknow ratings

Unknown ratings

Off-line Evaluation: Classification Metrics



Off-line Evaluation: Classification Metrics

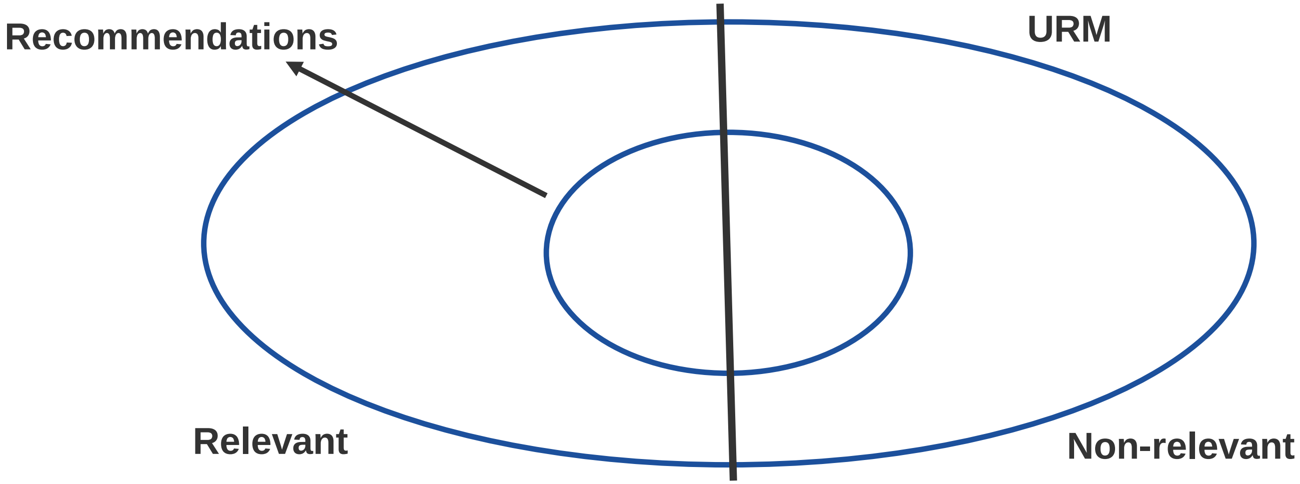


URM

Relevant

Non-relevant

Off-line Evaluation: Classification Metrics



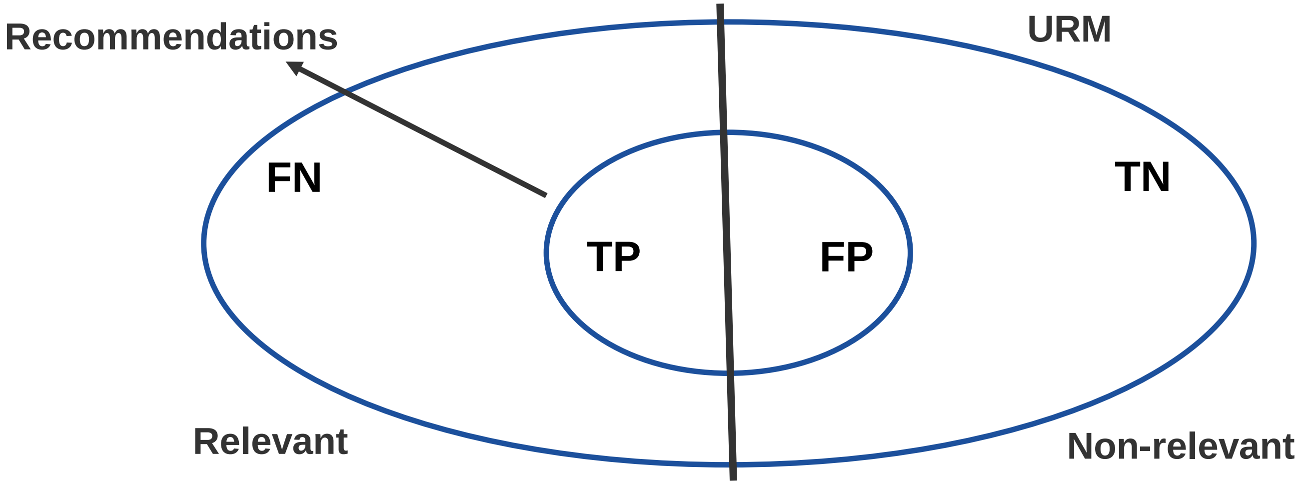
Recommendations

URM

Relevant

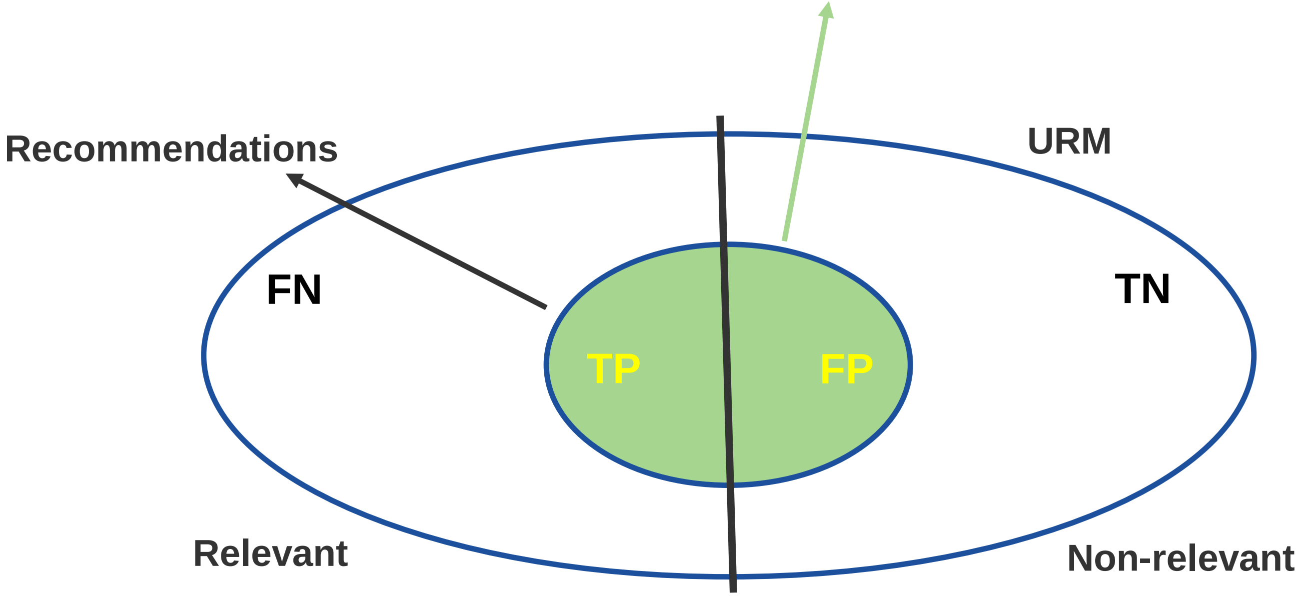
Non-relevant

Off-line Evaluation: Classification Metrics



Off-line Evaluation: Classification Metrics

$\mathrm{Recall}(K)$ $\quad K = \#\text{recommended items}$



Recommendations

URM

FN

TN

TP

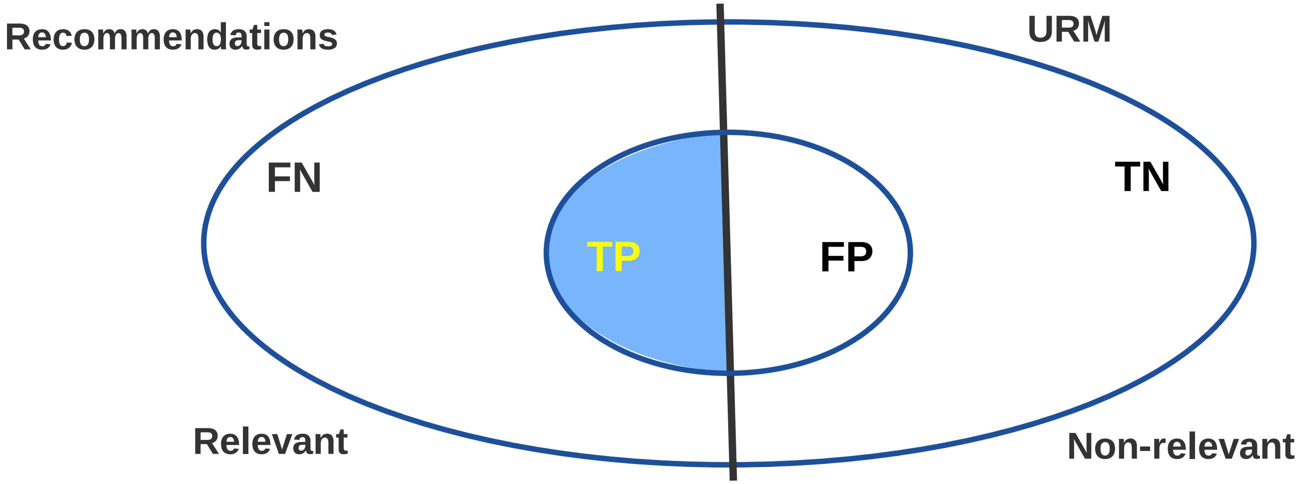
FP

Relevant

Non-relevant

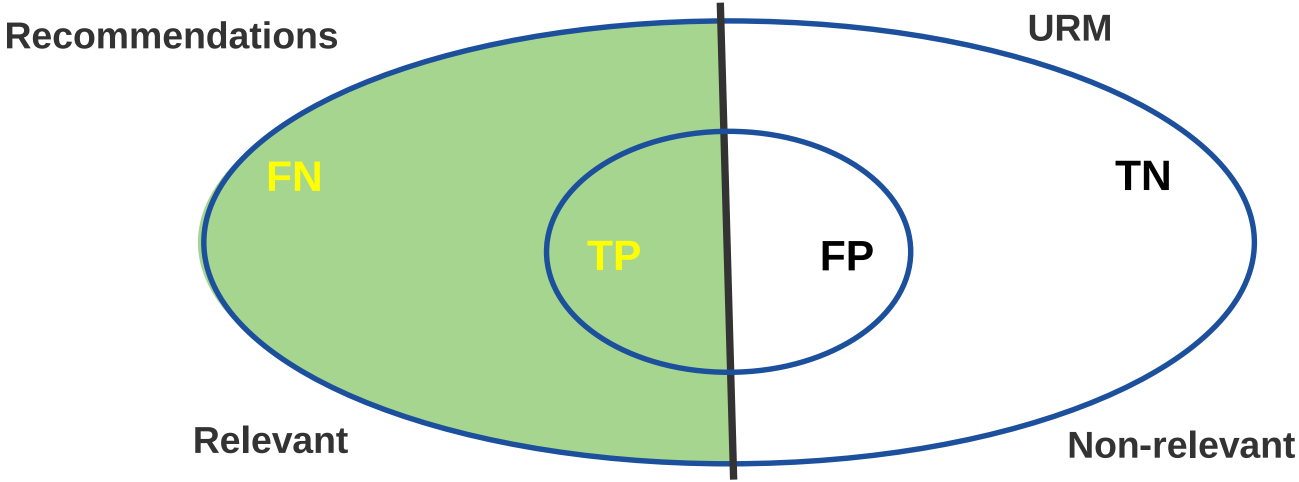
Off-line Evaluation: Classification Metrics

$$\text{Recall}(K) = \frac{\#\text{relevant recommended items}}{}$$



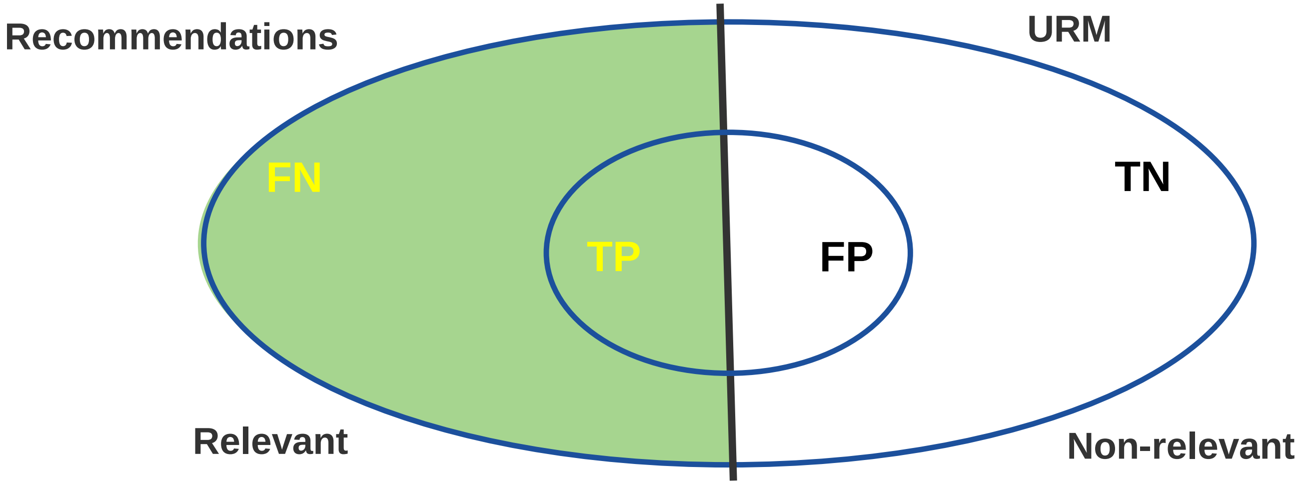
Off-line Evaluation: Classification Metrics

$$\text{Recall}(K) = \frac{\#\text{relevant recommended items}}{\#\text{tested relevant items}}$$



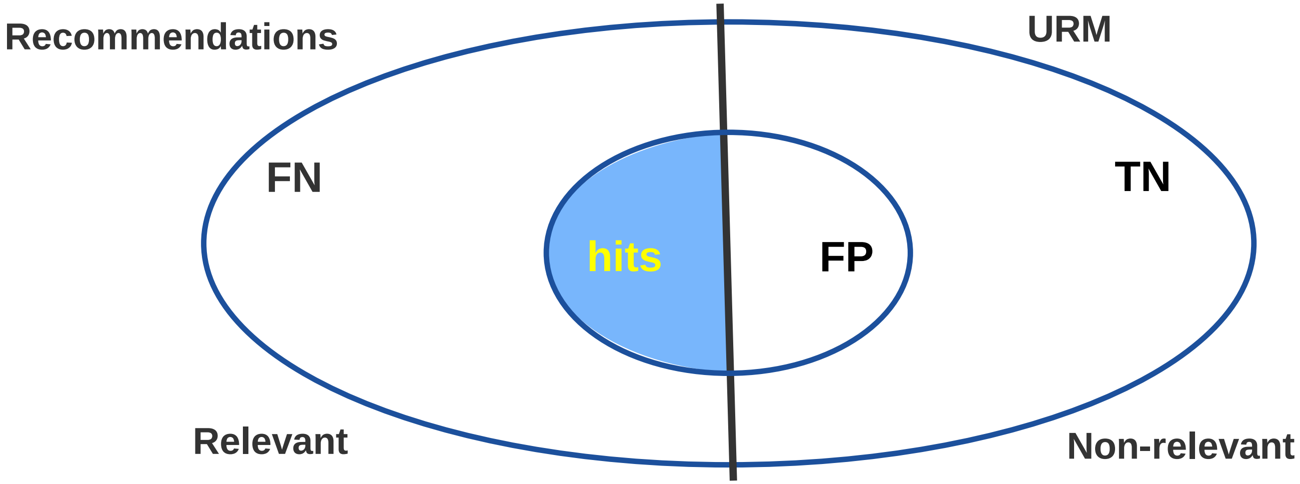
Off-line Evaluation: Classification Metrics

$$\mathrm{Recall}(K) = \frac{\#\text{relevant recommended items}}{\#\text{tested relevant items}} = \frac{\mathrm{TP}}{\mathrm{FN} + \mathrm{TP}}$$



Off-line Evaluation: Classification Metrics

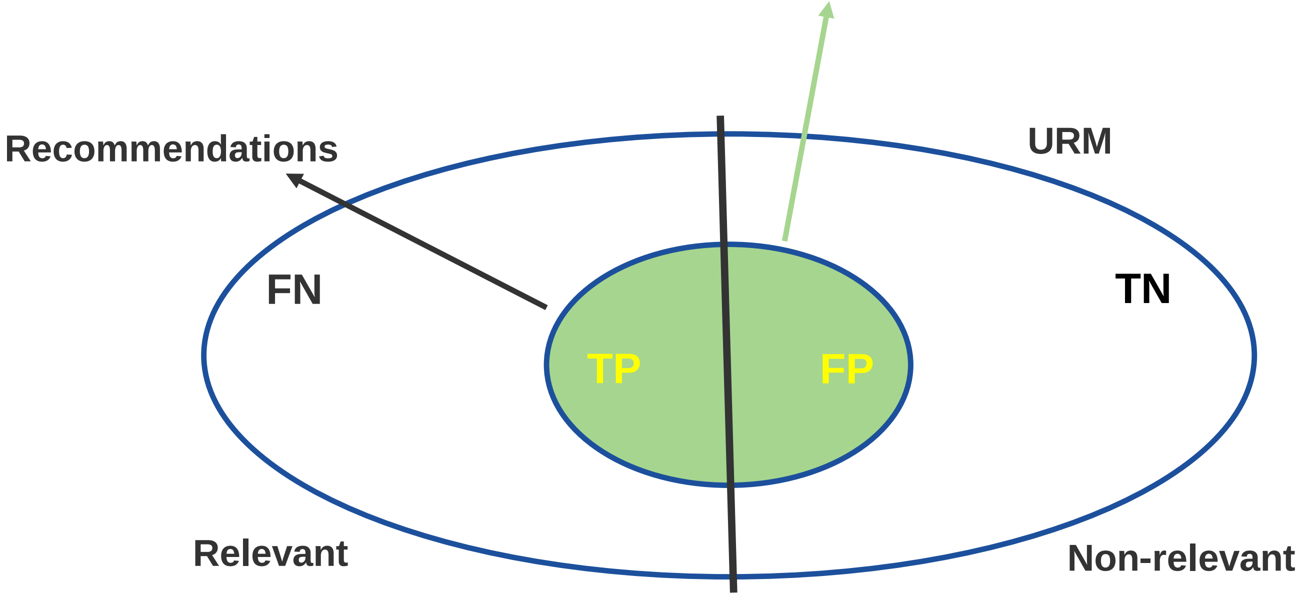
$$\text{Recall}(K) = \frac{\#\text{relevant recommended items}}{\#\text{tested relevant items}} = \frac{\#\text{hits}}{\text{FN} + \text{TP}}$$



Off-line Evaluation: Classification Metrics

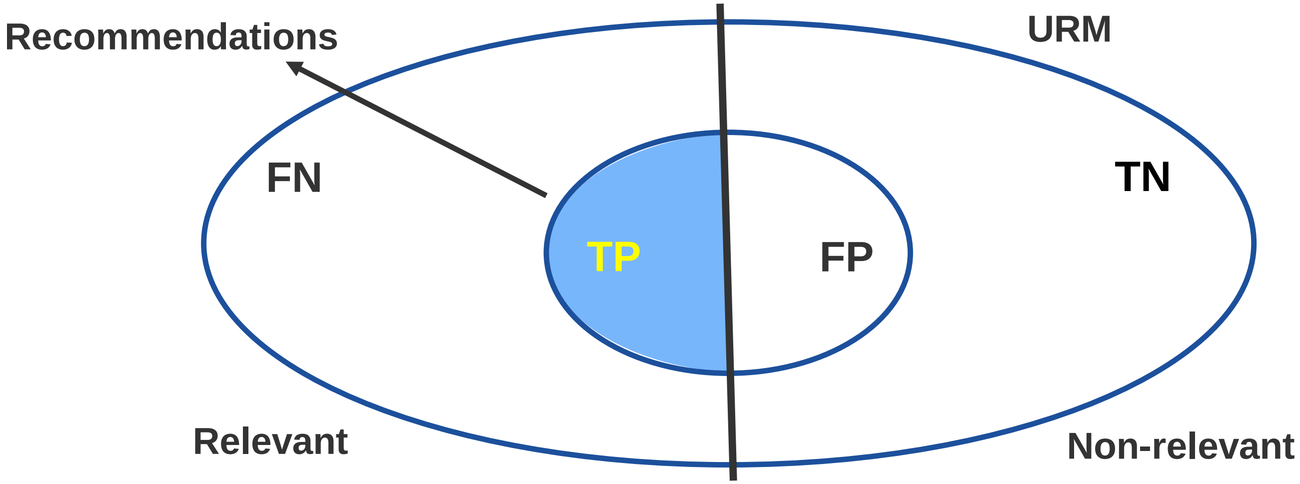
$$\text{Precision}(K) =$$

$K = \#\text{recommended items}$



Off-line Evaluation: Classification Metrics

$$\text{Precision}(K) = \frac{\#\text{relevant recommended items}}{}$$



Recommendations

URM

FN

TN

TP

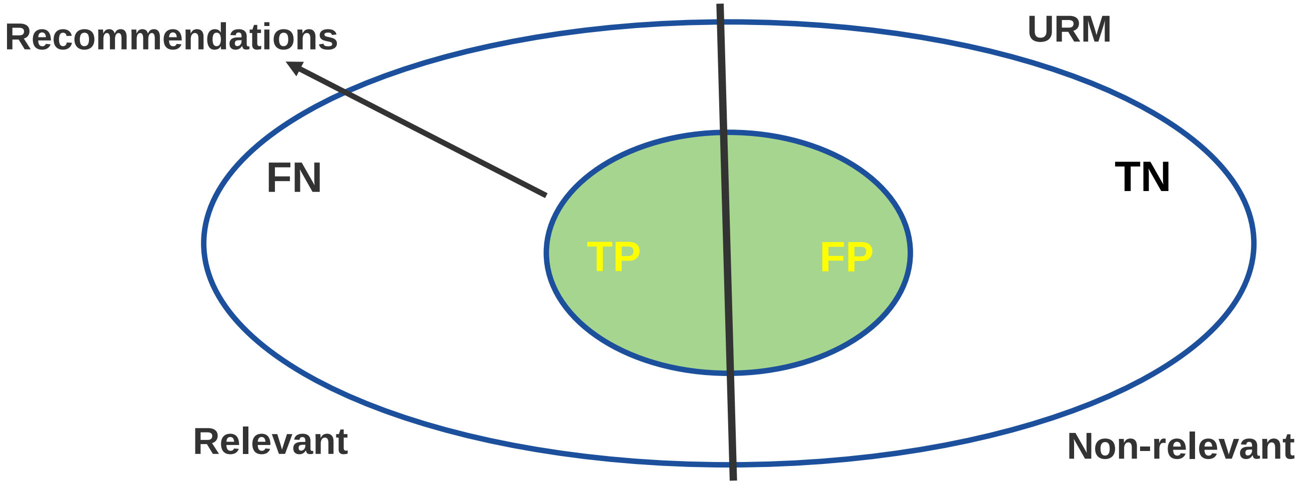
FP

Relevant

Non-relevant

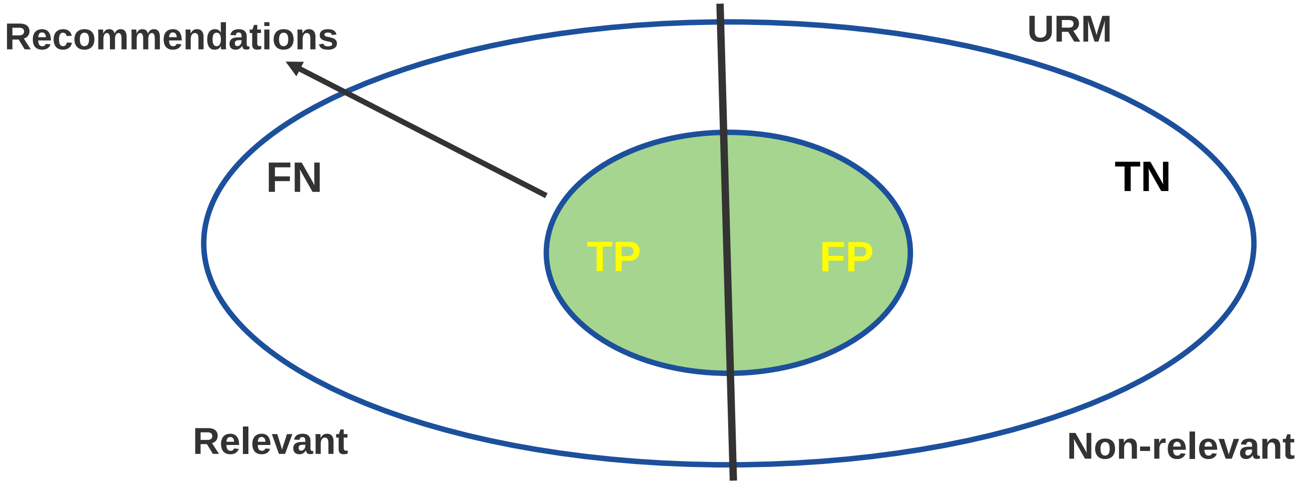
Off-line Evaluation: Classification Metrics

$$\text{Precision}(K) = \frac{\#\text{relevant recommended items}}{\#\text{all recommended items}}$$



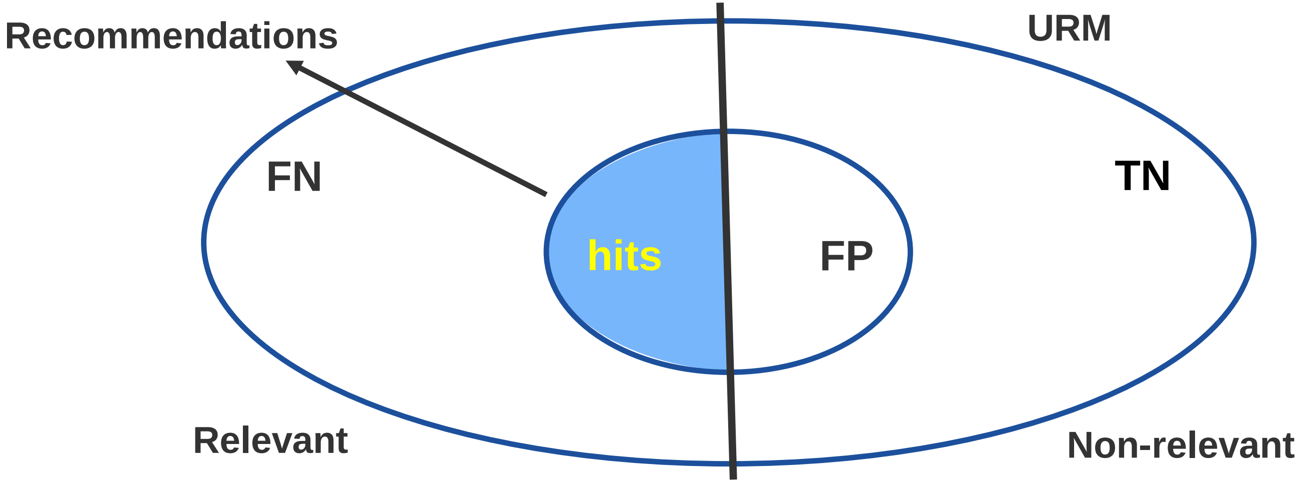
Off-line Evaluation: Classification Metrics

$$\text{Precision}(K) = \frac{\#\text{relevant recommended items}}{\#\text{all recommended items}} = \frac{\text{TP}}{\text{FP} + \text{TP}}$$



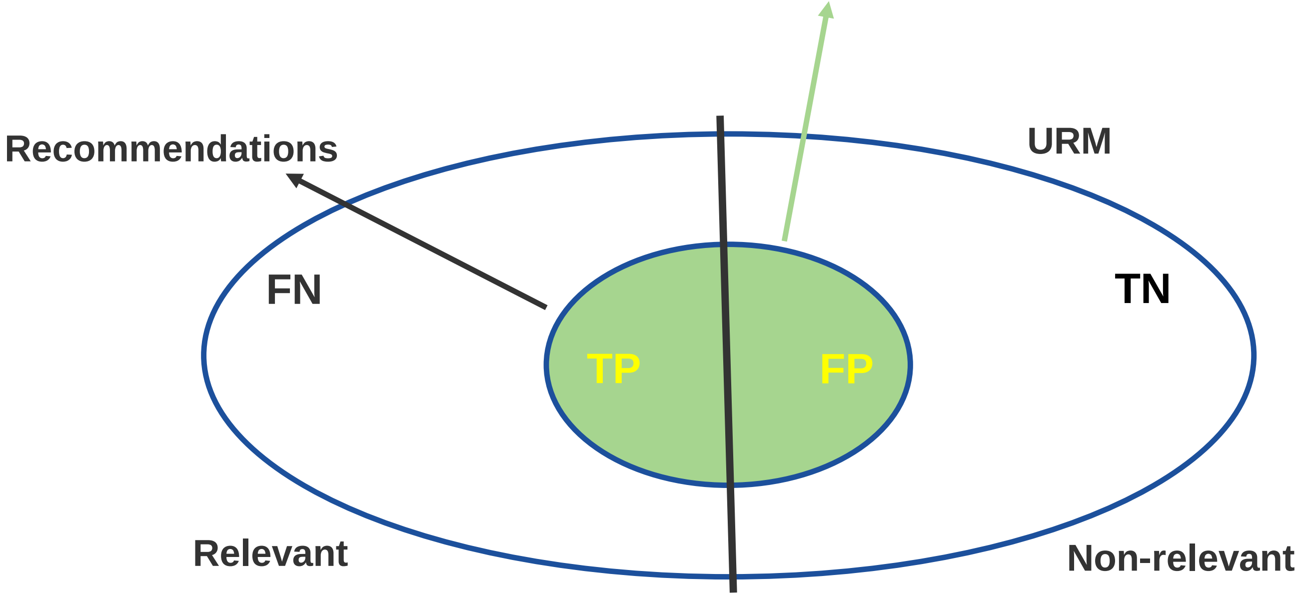
Off-line Evaluation: Classification Metrics

$$\text{Precision}(K) = \frac{\#\text{relevant recommended items}}{\#\text{all recommended items}} = \frac{\#\text{hits}}{\text{FP} + \text{TP}}$$



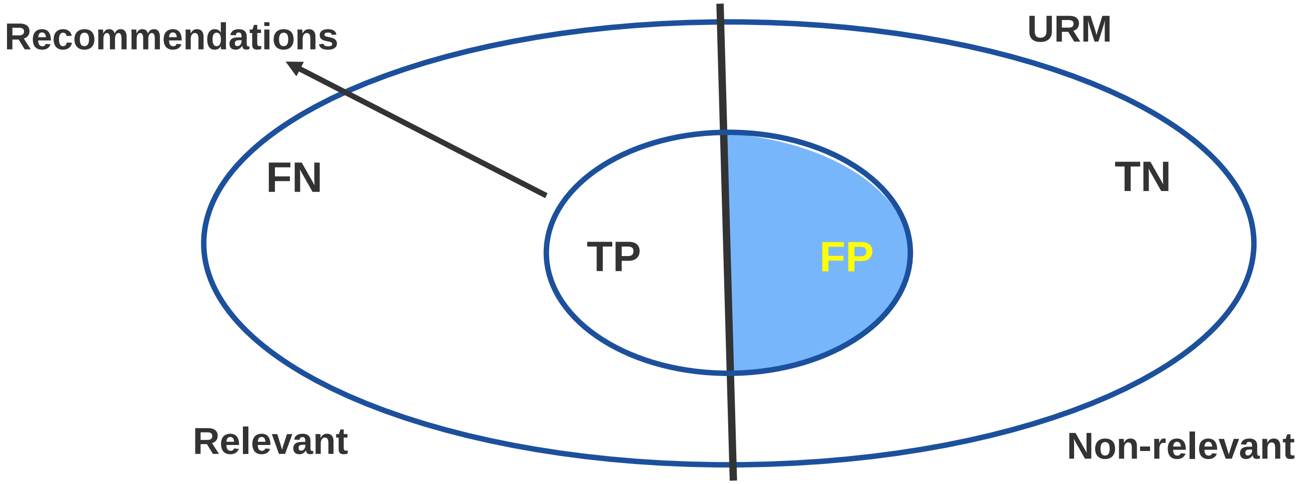
Off-line Evaluation: Classification Metrics

$\text{Fallout}(K) =$ $K = \#\text{recommended items}$



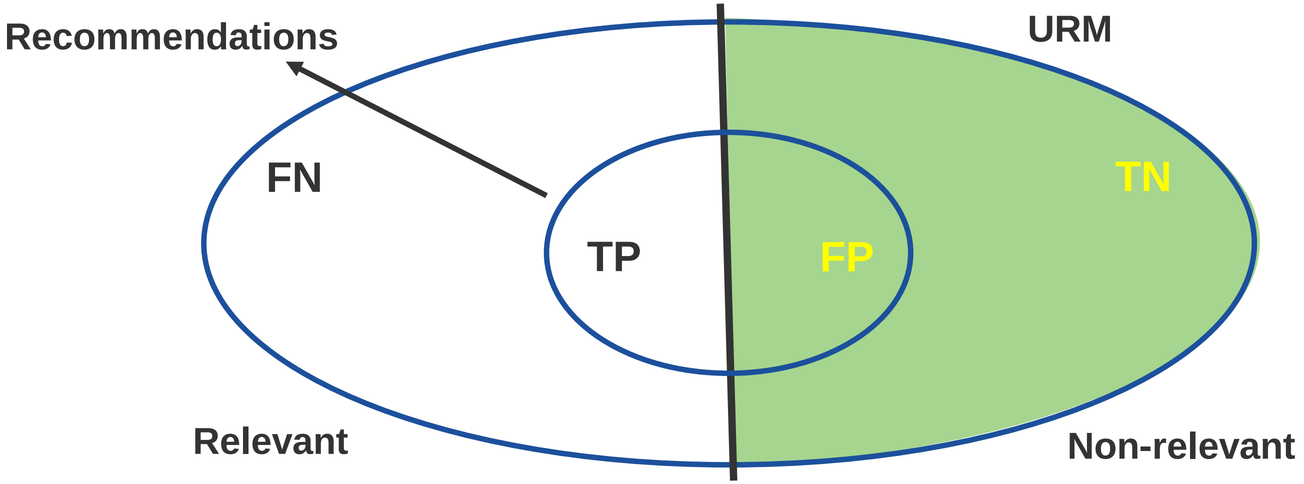
Off-line Evaluation: Classification Metrics

$$\text{Fallout}(K) = \frac{\#\text{non relevant recommended items}}{}$$



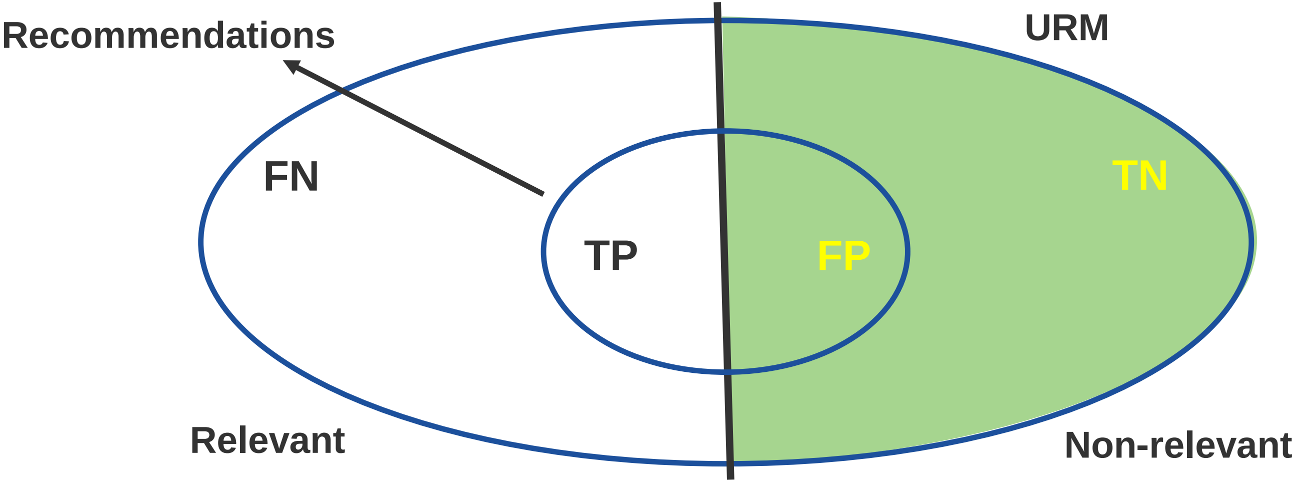
Off-line Evaluation: Classification Metrics

$$\text{Fallout}(K) = \frac{\#\text{non relevant recommended items}}{\#\text{all non relevant items}}$$



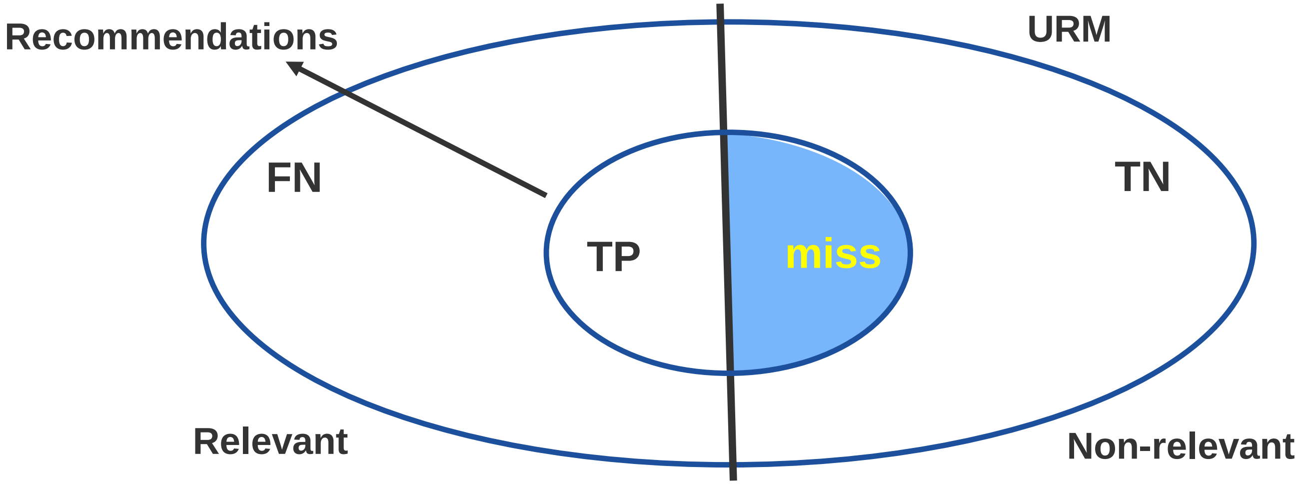
Off-line Evaluation: Classification Metrics

$$\text{Fallout}(K) = \frac{\text{\#non relevant recommended items}}{\text{\#all non relevant items}} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

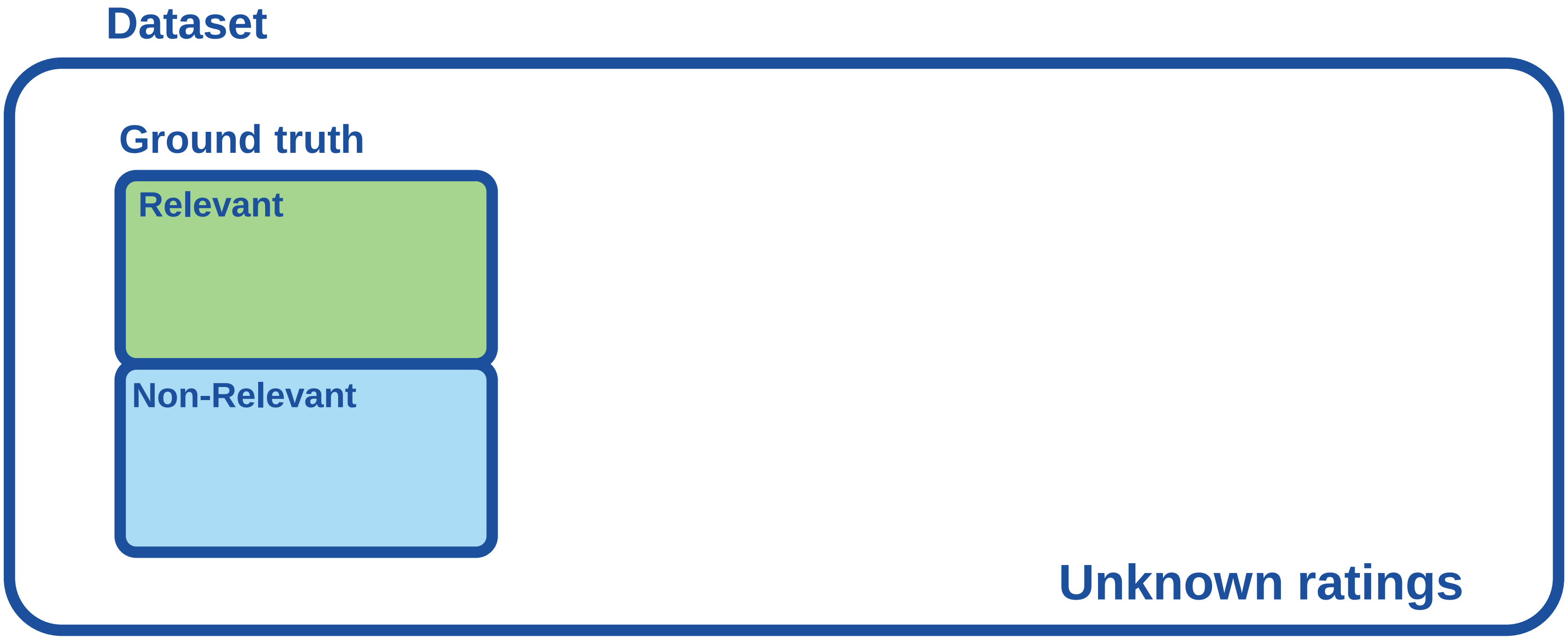


Off-line Evaluation: Classification Metrics

$$\text{Fallout}(K) = \frac{\#\text{non relevant recommended items}}{\#\text{all non relevant items}} = \frac{\#\text{miss}}{\text{FP} + \text{TN}}$$



Off-line Evaluation: Classification Metrics



Dataset

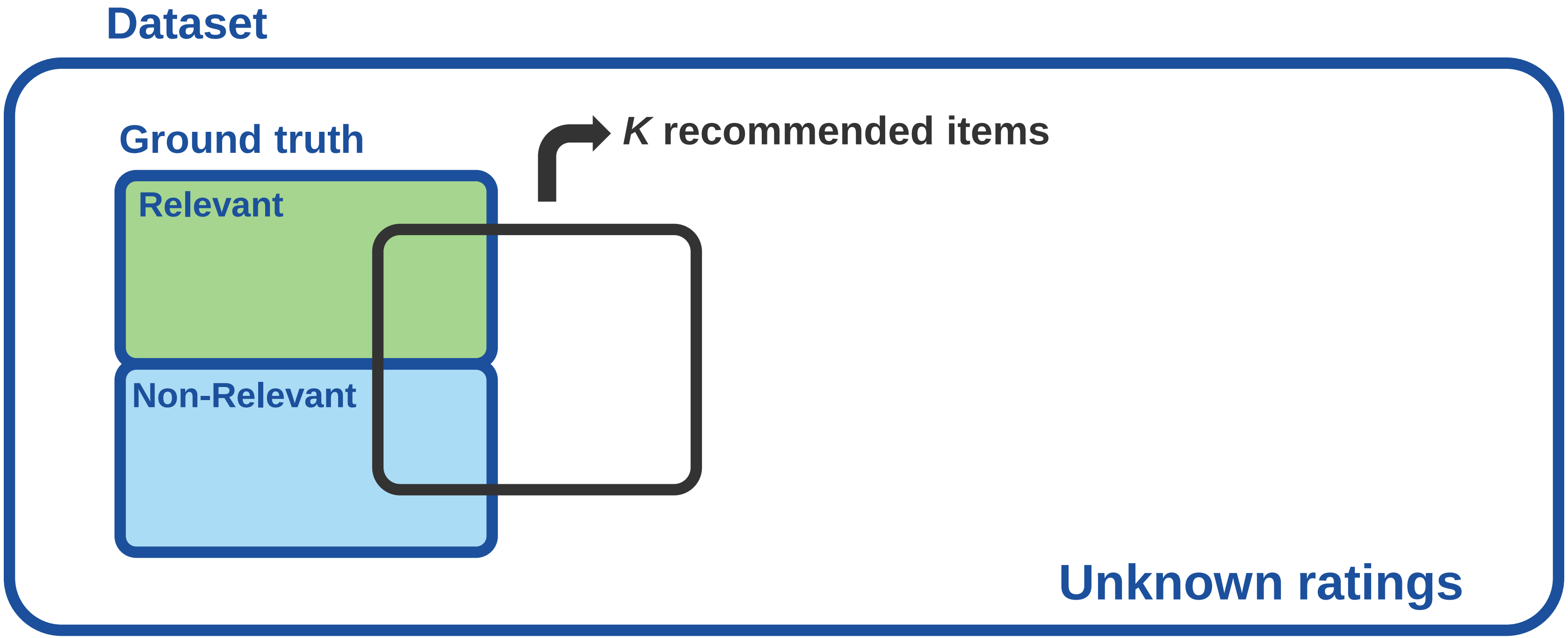
Ground truth

Relevant

Non-Relevant

Unknown ratings

Off-line Evaluation: Classification Metrics



Dataset

Ground truth

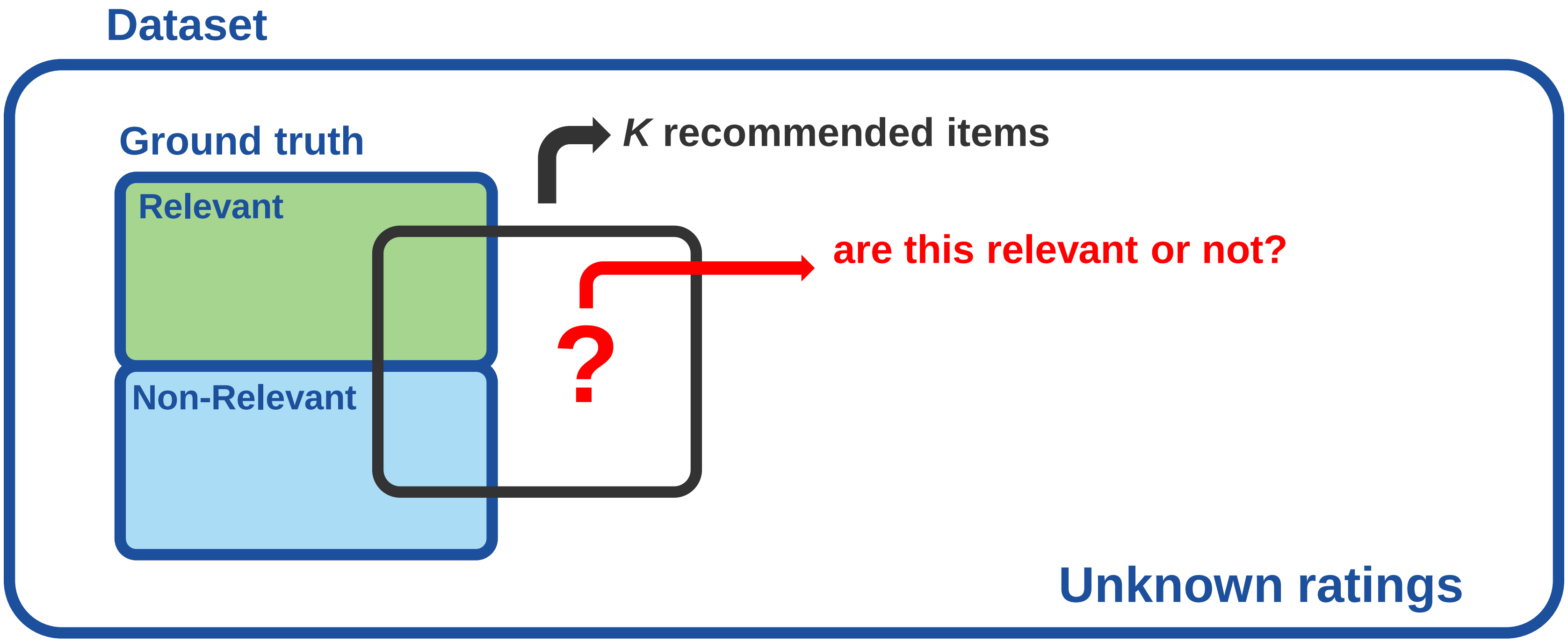
Relevant

Non-Relevant

***K* recommended items**

Unknown ratings

Off-line Evaluation: Classification Metrics



Dataset

Ground truth

Relevant

Non-Relevant

***K* recommended items**

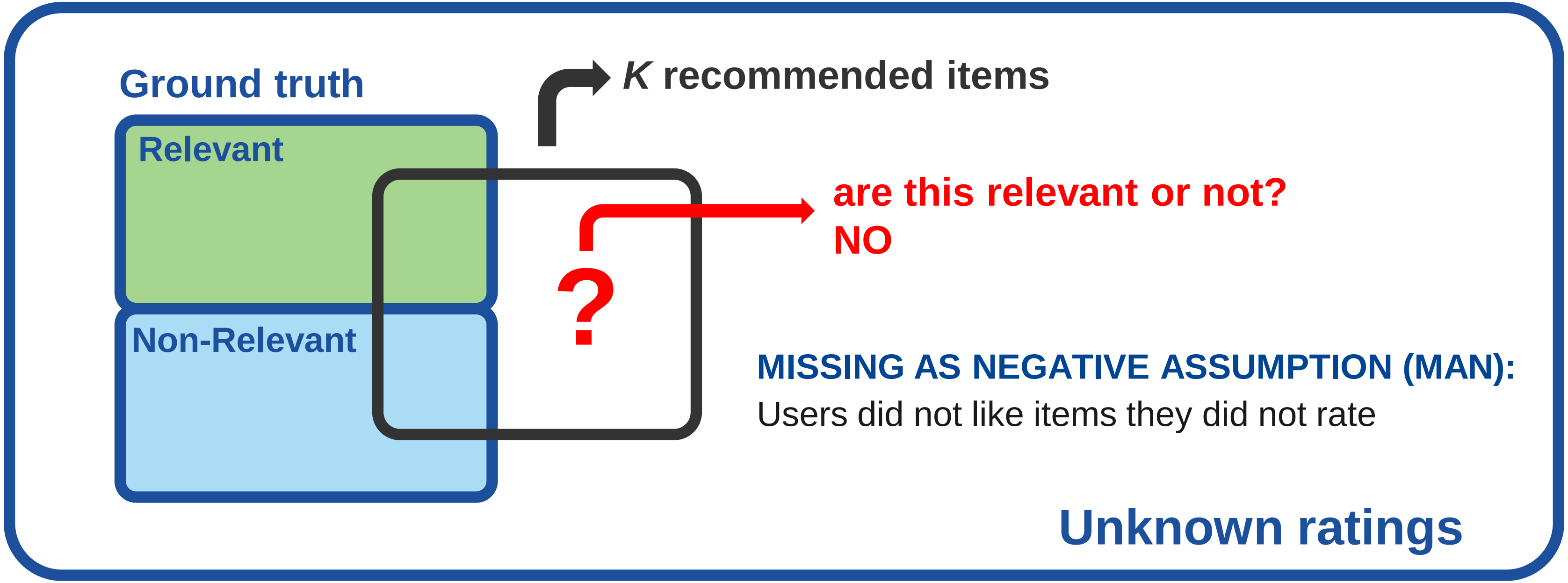
?

are this relevant or not?

Unknown ratings

Off-line Evaluation: Classification Metrics

Dataset



Ground truth

Relevant

Non-Relevant

***K* recommended items**

?

are this relevant or not?
NO

MISSING AS NEGATIVE ASSUMPTION (MAN):
Users did not like items they did not rate

Unknown ratings

Precision

All-Missing-As-Negative (AMAN) hypothesis

- all missing ratings are irrelevant
- underestimate the true precision computed on the (unknown) complete data

Harald Steck, *Training and testing of RSs on data missing not at random*. In KDD '10

Recall

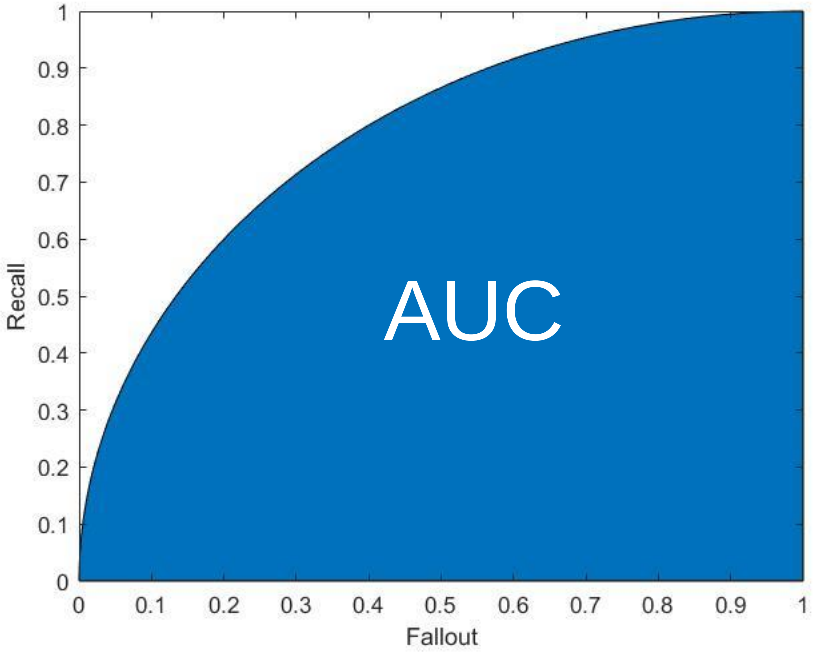
Missing-Not-At-Random (MNAR) hypothesis

- non-relevant ratings are missing with a higher probability than relevant ratings
- (nearly) unbiased estimate of recall on the (unknown) complete data
- much milder than assuming that

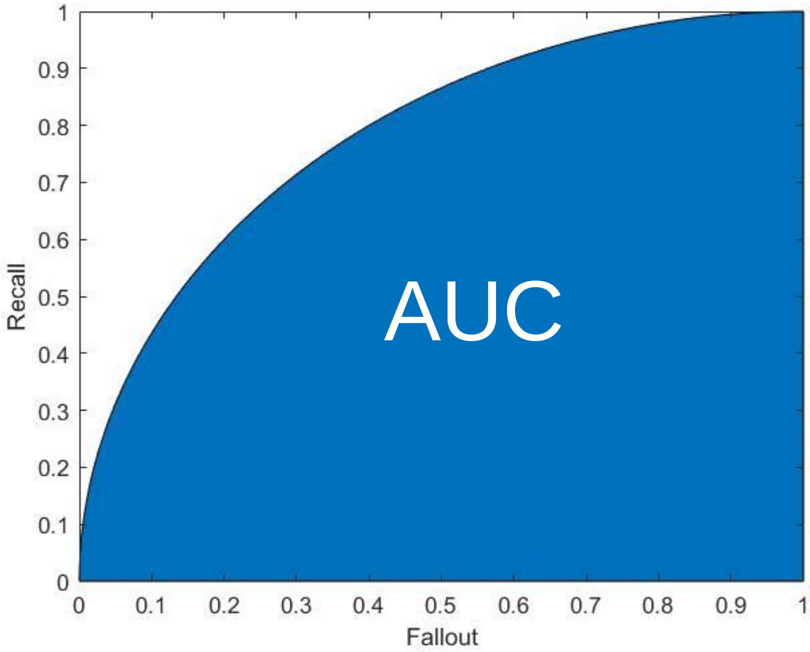
all the ratings are missing at random (MAE and RMSE)
all missing ratings are irrelevant (Precision)

Ranking Metrics

ROC curve (area under curve)

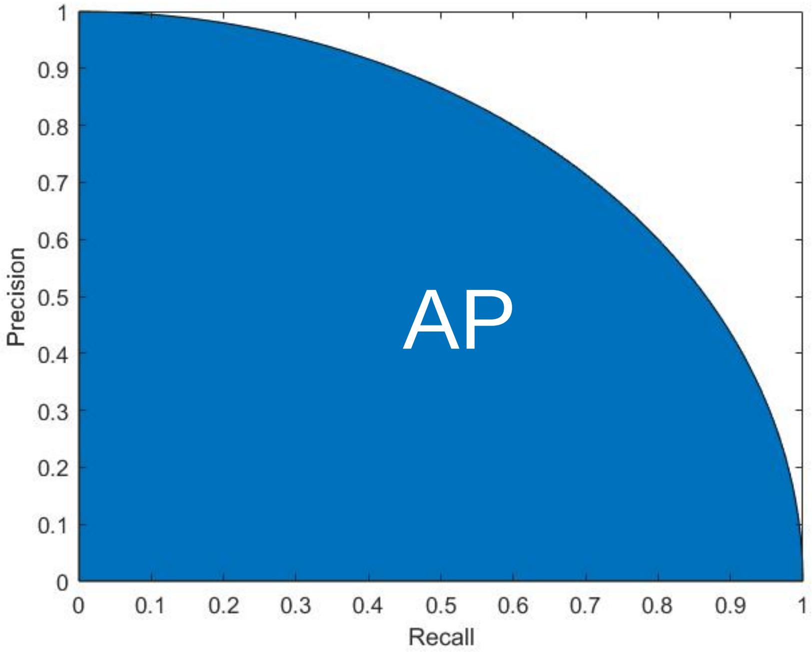


ROC curve (area under curve)

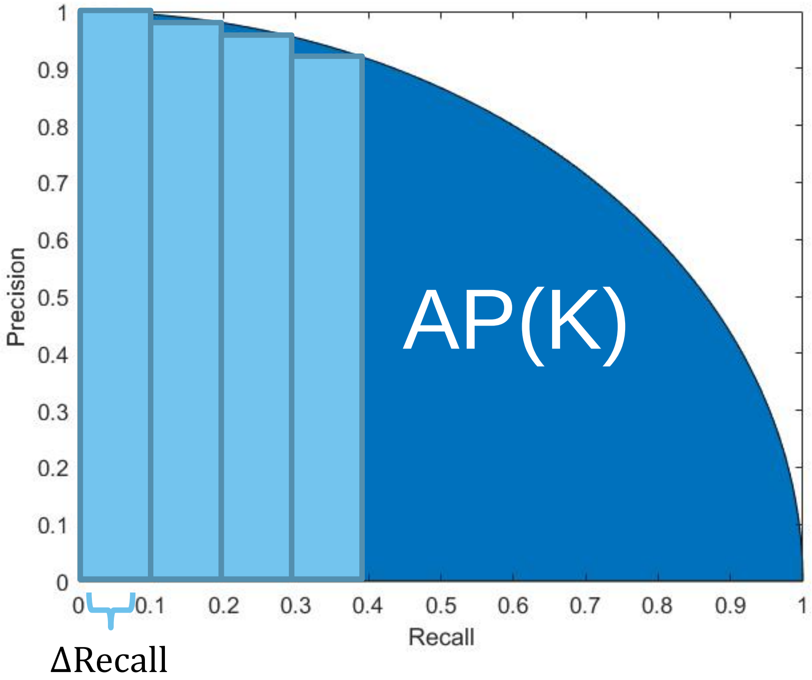


$$AUC = \Sigma_k \text{Recall}(k) * \Delta\text{Fallout}$$

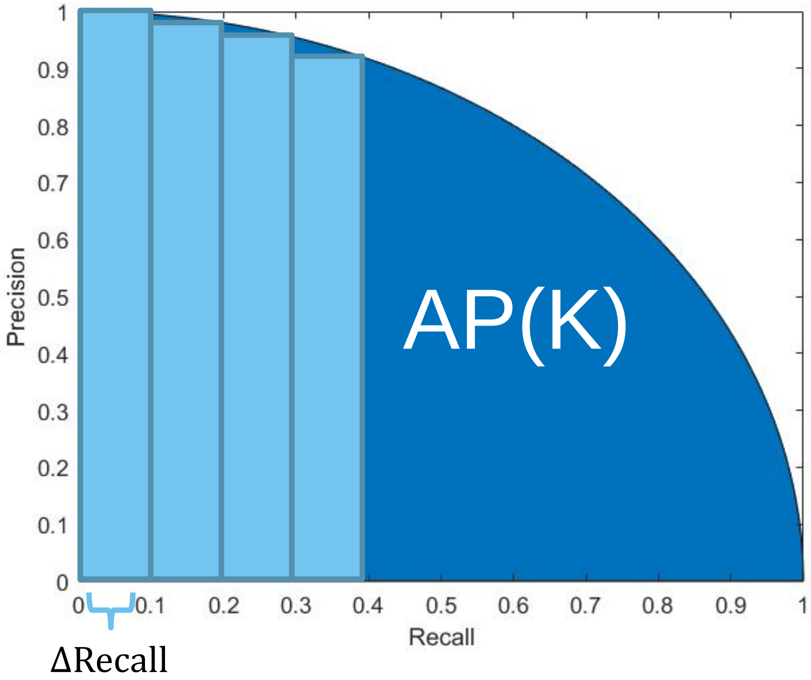
Mean Average Precision



Mean Average Precision

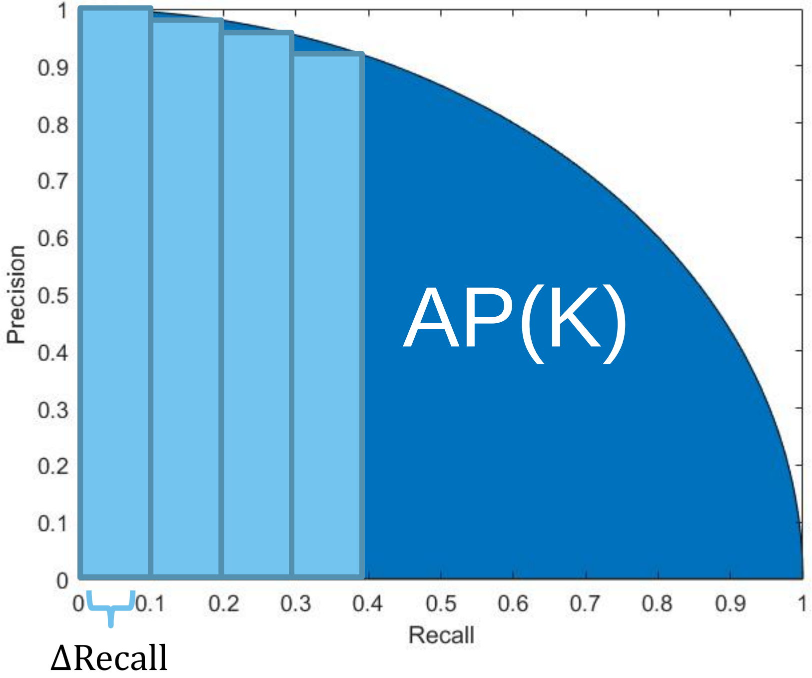


Mean Average Precision



$$AP = \Sigma_k \mathrm{Precision}(k) * \Delta\mathrm{Recall}$$

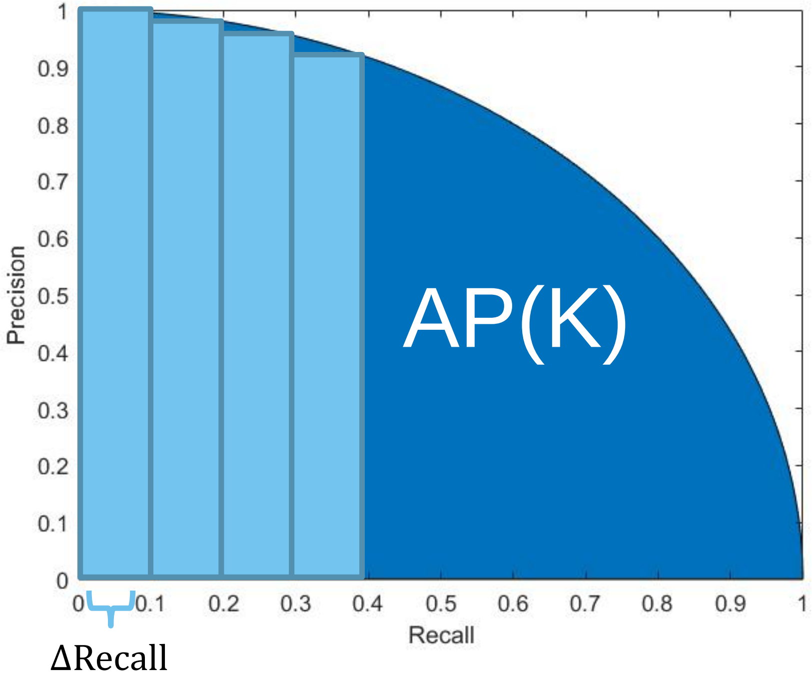
Mean Average Precision



$$AP = \Sigma_k \text{Precision}(k) * \Delta\text{Recall}$$

$$\Delta\text{Recall} = \text{Recall}(k) - \text{Recall}(k-1)$$

Mean Average Precision



$$AP = \Sigma_k \text{Precision}(k) * \Delta\text{Recall}$$

$$\text{MAP} = \frac{\Sigma_u AP_u}{\#\text{users}}$$

Average Reciprocal Hit-Rank

Weighted version of recall

$$ARHR = \frac{\Sigma_i \frac{1}{\text{rank(i)}}}{\#\text{tested relevant items}}$$

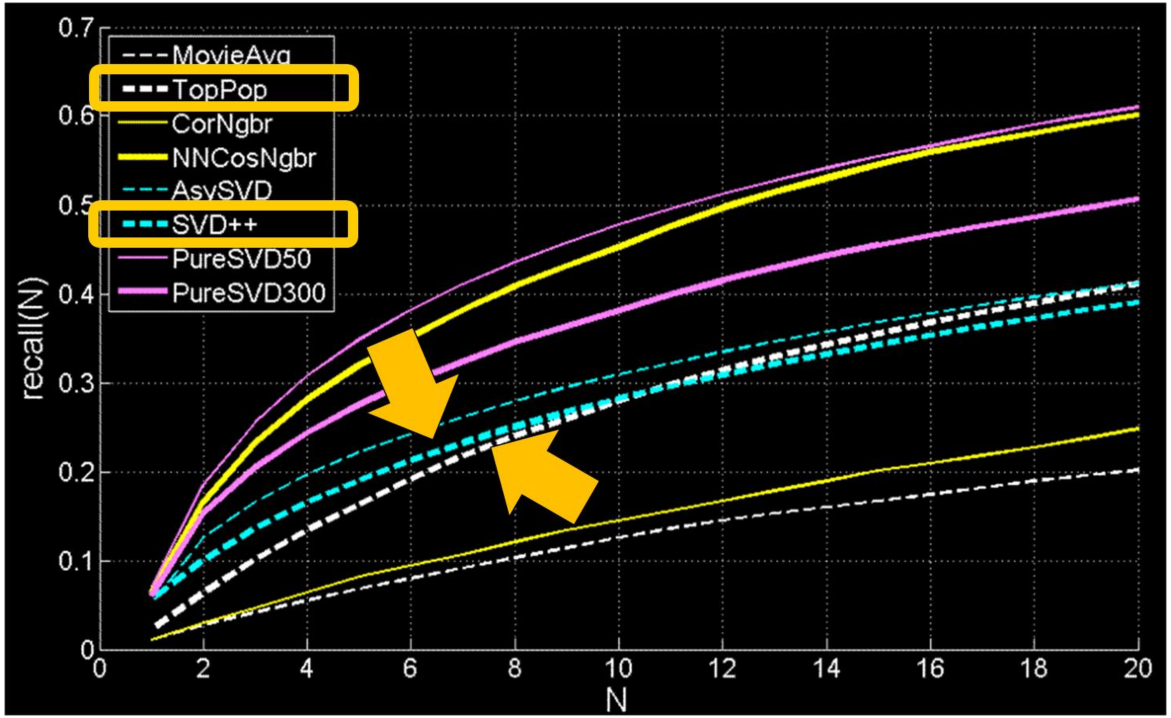
Average Reciprocal Hit-Rank

Weighted version of recall

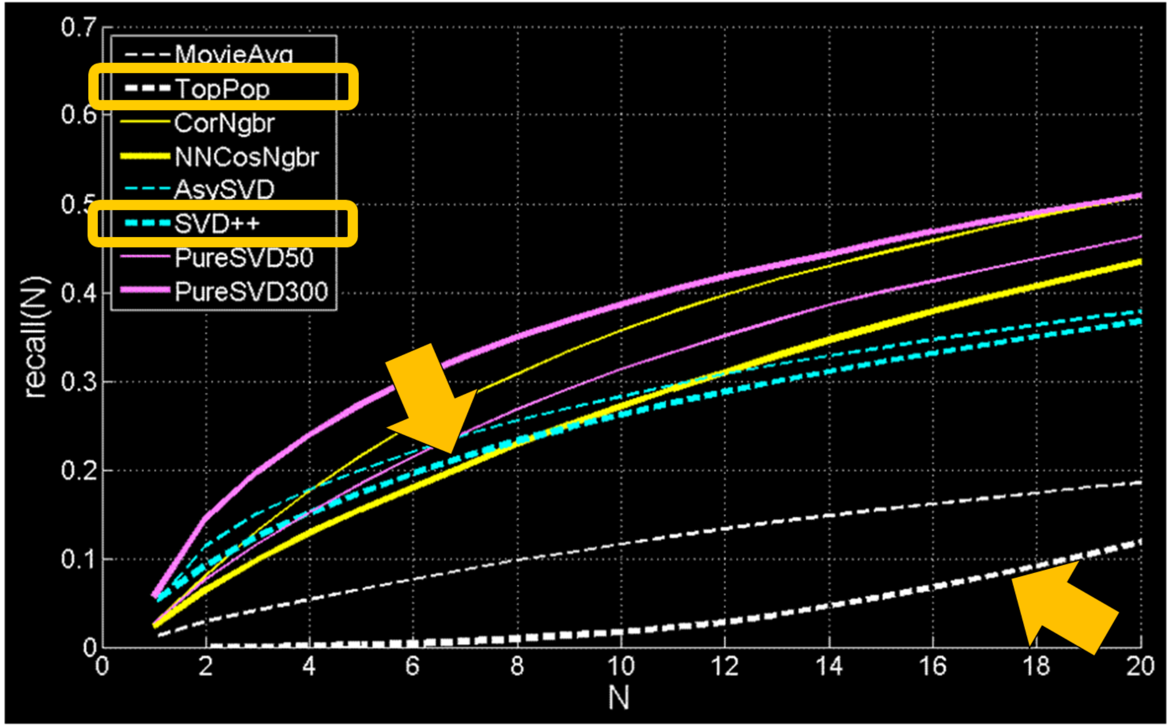
$$ARHR = \frac{\Sigma_i \frac{1}{\text{rank(i)}}}{\#\text{tested relevant items}}$$

i: **relevant** item recommended to the user
rank(i): position of item *i* in the list of recommendations

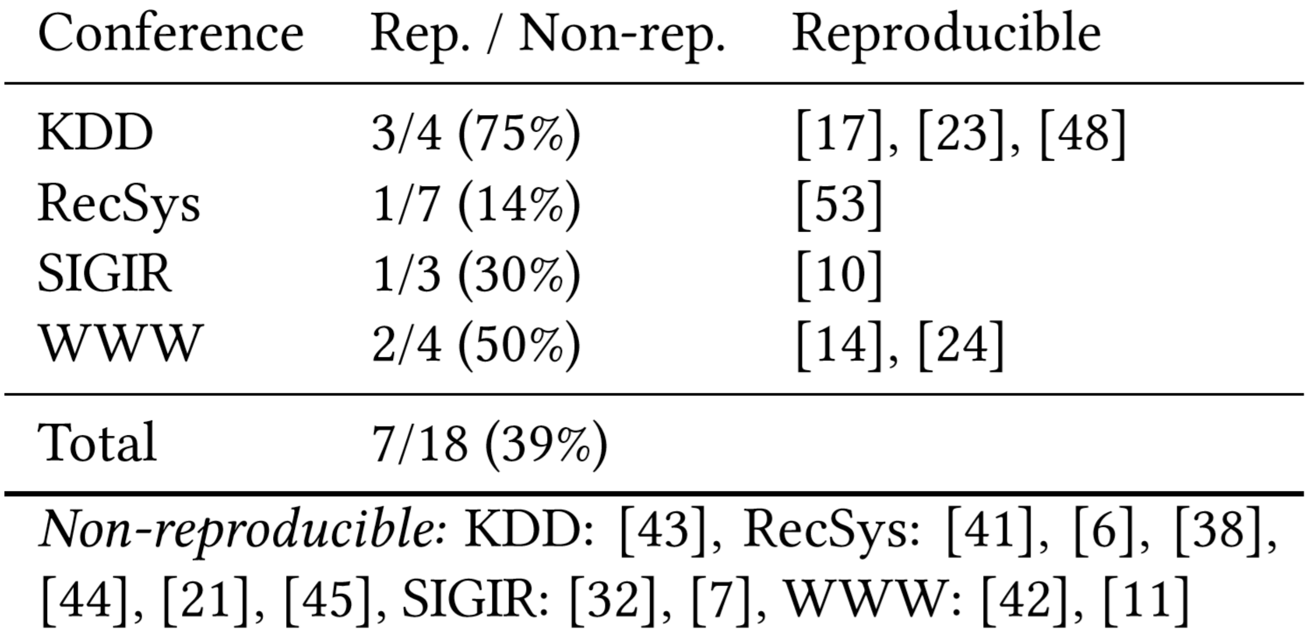
Netflix dataset: recall



Netflix dataset: recall on long tail



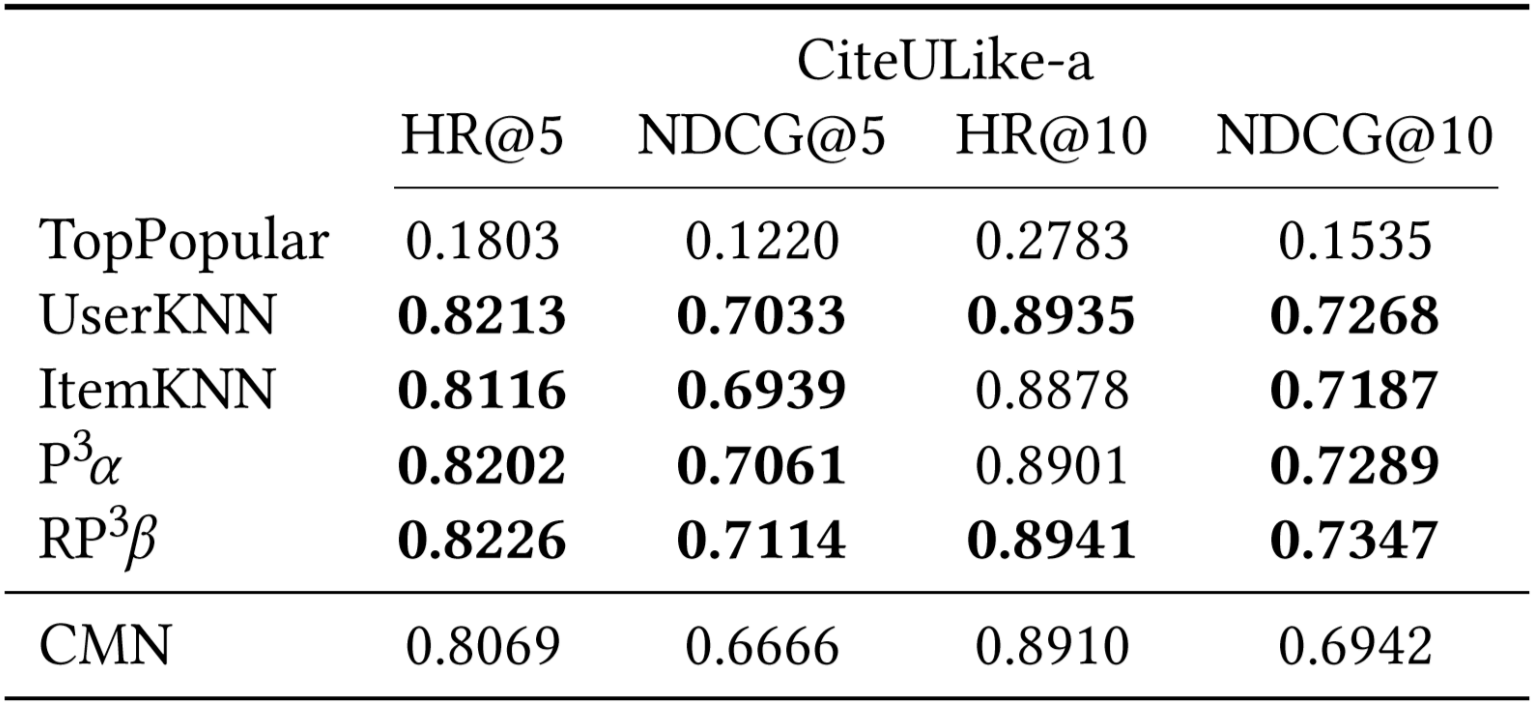
Evaluation: are we really making much progress?



Conference	Rep. / Non-rep.	Reproducible
KDD	3/4 (75%)	[17], [23], [48]
RecSys	1/7 (14%)	[53]
SIGIR	1/3 (30%)	[10]
WWW	2/4 (50%)	[14], [24]
Total	7/18 (39%)	

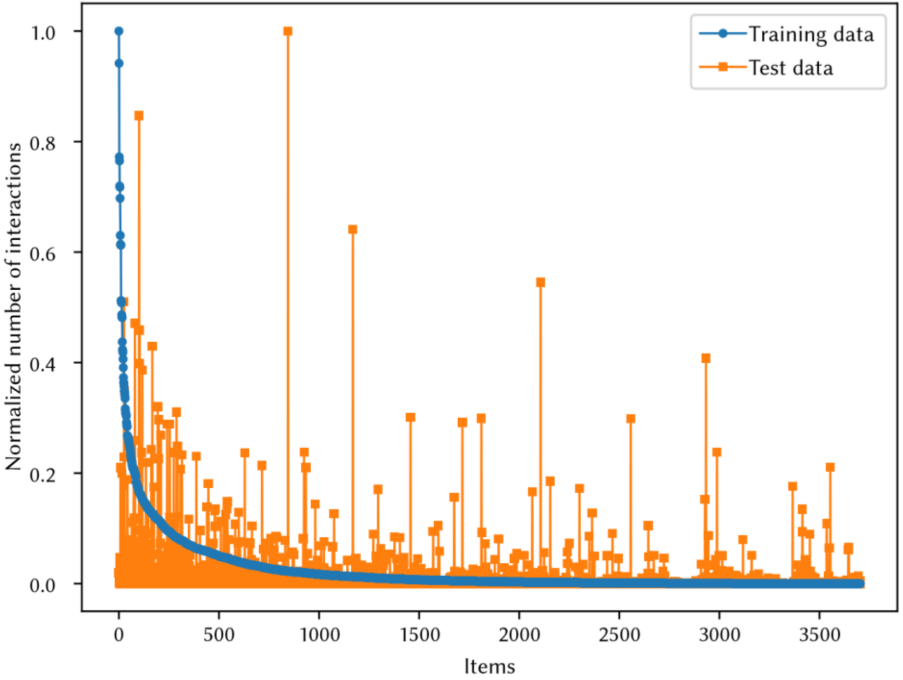
Non-reproducible: KDD: [43], RecSys: [41], [6], [38], [44], [21], [45], SIGIR: [32], [7], WWW: [42], [11]

Evaluation: are we really making much progress?



	CiteULike-a			
	HR@5	NDCG@5	HR@10	NDCG@10
TopPopular	0.1803	0.1220	0.2783	0.1535
UserKNN	**0.8213**	**0.7033**	**0.8935**	**0.7268**
ItemKNN	**0.8116**	**0.6939**	0.8878	**0.7187**
$P^3\alpha$	**0.8202**	**0.7061**	0.8901	**0.7289**
$RP^3\beta$	**0.8226**	**0.7114**	**0.8941**	**0.7347**
CMN	0.8069	0.6666	0.8910	0.6942

Evaluation: are we really making much progress?



Evaluating
Non Accuracy Metrics

Measuring Diversity

$$\text{diversity} = \frac{\Sigma_{i,j} 1 - \text{similarity}(i,j)}{N(N-1)}$$

Measuring Diversity

$$\text{diversity} = \frac{\Sigma_{i,j} 1 - \text{similarity}(i,j)}{N(N-1)}$$

N: total number of items
i,j: considered items

Measuring Novelty

$$\text{Novelty} = \frac{\#\text{relevant and } \textbf{unknown} \text{ recommended items}}{\#\text{recommended relevant items}}$$

Measuring Novelty

$$\text{Novelty} = \frac{\#\text{relevant and } \textbf{unknown} \text{ recommended items}}{\#\text{recommended relevant items}}$$

$$novelty \ \approx 1/popularity$$

Measuring Novelty

$$\text{Novelty} = \frac{\#\text{relevant and } \textbf{unknown} \text{ recommended items}}{\#\text{recommended relevant items}}$$

$$novelty \approx 1/popularity$$

$$\text{novelty} = \frac{\Sigma_{i\in\text{hits}} \log_2\left(\frac{1}{\text{popularity}(i)}\right)}{\#\text{hits}}$$

Measuring Novelty

$$\text{Novelty} = \frac{\#\text{relevant and } \textbf{unknown} \text{ recommended items}}{\#\text{recommended relevant items}}$$

$$novelty \ \approx 1/popularity$$

$$\text{novelty} = \frac{\Sigma_{i \in \text{hits}} \log_2\left(\frac{1}{\text{popularity}(i)}\right)}{\#\text{hits}}$$

$popularity(i)$ =
% of users who rated item *i*

Paolo Cremonesi

paolo.cremonesi@polimi.it