The ACM Conference Series on Recommender Systems

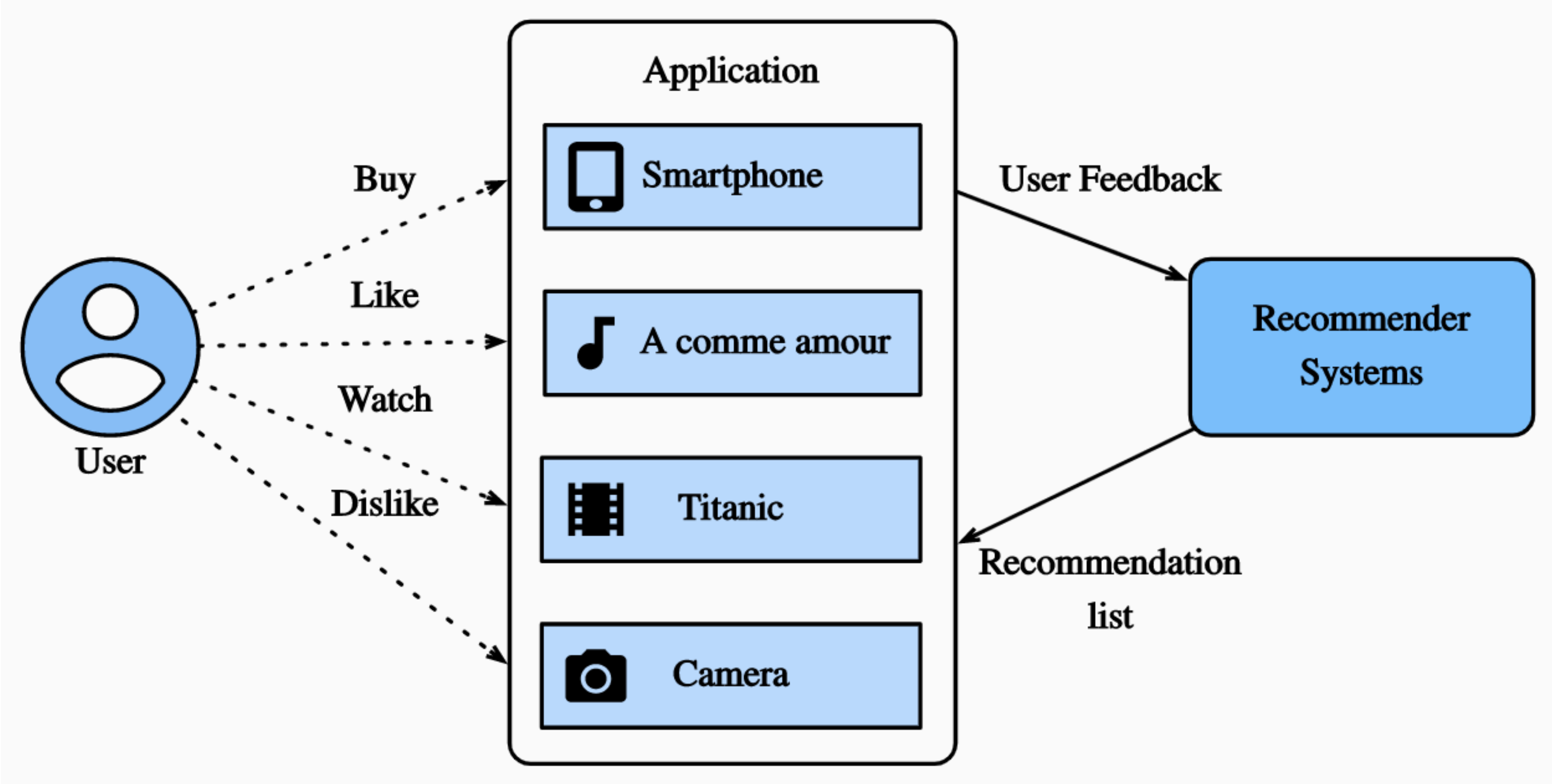
NANYANG TECHNOLOGICAL UNIVERSITY

On Challenges of Evaluating Recommender Systems in Offline Setting

Dr. Aixin Sun

NTU Singapore

Recommender System



Source: https://d2l.ai/chapter_recommender-systems/recsys-intro.html

Outline

- Recommender system basics
  - Recommender system evaluation
  - Commonly used metrics in academic research and practice
- Challenges in computing the offline metrics
  - Data partition schemes in RecSys experiments using offline datasets
  - Data leakage due to not maintaining global timeline
  - The impact on understanding the RecSys research problem
- Criticism on RecSys from evaluation perspective
  - The counter-intuitive observations
  - The common pitfalls in evaluating RecSys
- More practical evaluations
  - The meaning of fair comparison
  - The observation of global timeline

Recommender Systems: Examples

- Products on e-commerce websites
- Online content
  - Video
  - Music
  - News
- Advertisement
- Social media

DIAMOND SUPPORTER

HUAWEI

TikTok

PLATINUM SUPPORTER

NETFLIX RESEARCH

Coolita

amazon | science

GOLD SUPPORTER

Google

SILVER SUPPORTER

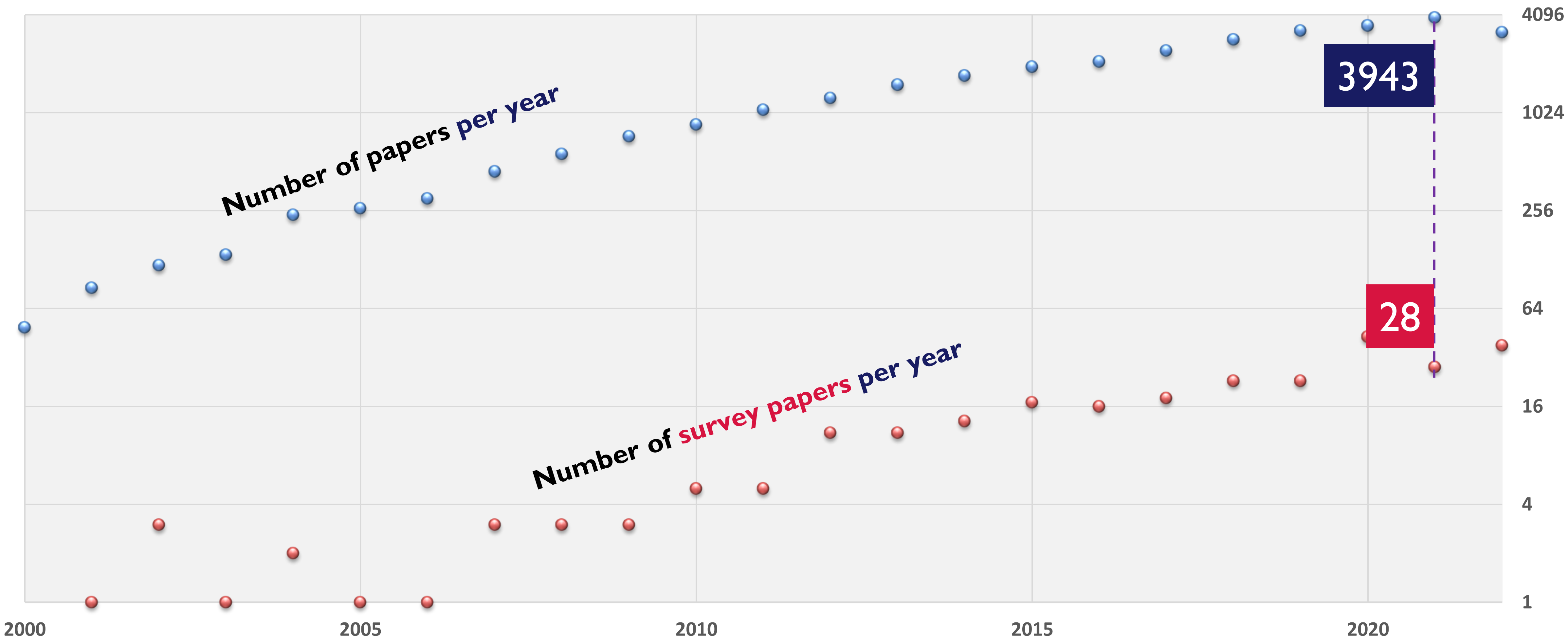
SiriusXM

BRONZE SUPPORTER

CHALLENGE SPONSOR

ShareChat

RecSys is a problem-rich research area



https://dblp.org/search/publ?q=recommend https://dblp.org/search/publ?q=recommend%20survey

RecSys Evaluation

Evaluating Recommender Systems: Survey and Framework

EVA ZANGERLE, Universität Innsbruck, Austria
CHRISTINE BAUER, Utrecht University, The Netherlands

The comprehensive evaluation of the performance of a recommender system is a complex endeavor: many facets need to be considered in configuring an adequate and effective evaluation setting. Such facets include, for instance, defining the specific goals of the evaluation, choosing an evaluation method, underlying data, and suitable evaluation metrics. In this article, we consolidate and systematically organize this dispersed knowledge on recommender systems evaluation. We introduce the Framework for Evaluating Recommender systems (FEVR), which we derive from the discourse on recommender systems evaluation. In FEVR, we categorize the evaluation space of recommender systems evaluation. We postulate that the comprehensive evaluation of a recommender system frequently requires considering multiple facets and perspectives in the evaluation. The FEVR framework provides a structured foundation to adopt adequate evaluation configurations that encompass this required multi-facetedness and provides the basis to advance in the field. We outline and discuss the challenges of a comprehensive evaluation of recommender systems and provide an outlook on what we need to embrace and do to move forward as a research community.

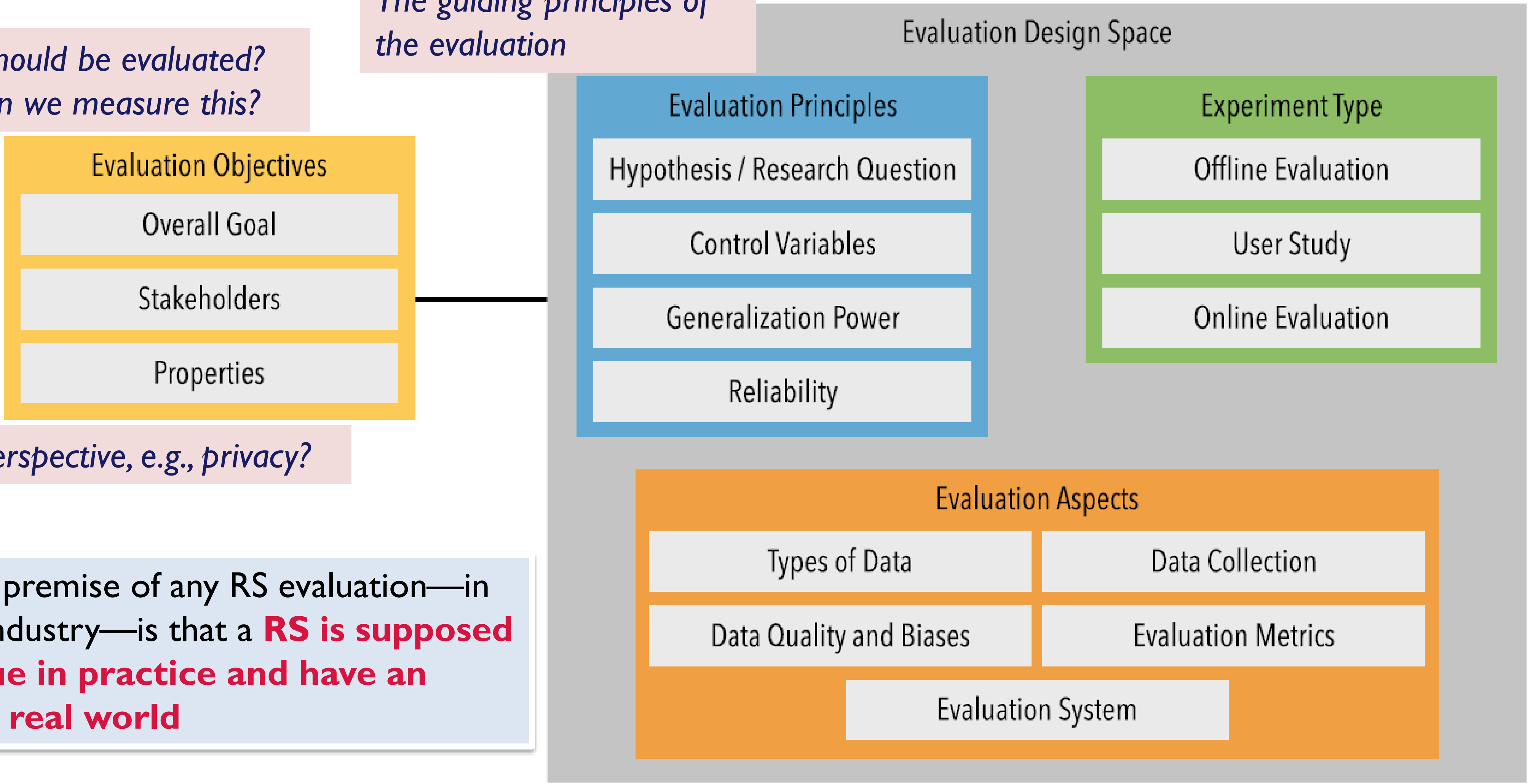
- The comprehensive evaluation of the performance of a recommender system is a complex endeavor
  - Defining the **specific goals** of the evaluation
  - Choosing
    - Evaluation method
    - Underlying data
    - Suitable evaluation metrics

- **System-centric**: the evaluation of algorithmic aspects, e.g., the predictive accuracy, revenue, CTR
- **User-centric**: how users perceive its quality or the user experience when interacting with the RS.

Framework for evaluating recommender systems (FEVR)

What should be evaluated?
How can we measure this?

The guiding principles of the evaluation



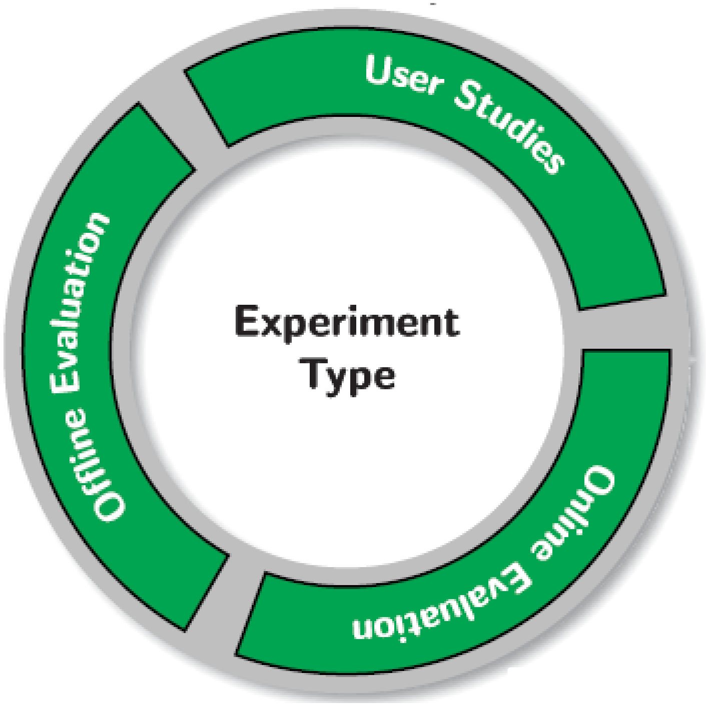
Which perspective, e.g., privacy?

The underlying premise of any RS evaluation—in academia and industry—is that a **RS is supposed to create value in practice and have an impact in the real world**

Eva Zangerle and Christine Bauer. 2022. Evaluating Recommender Systems: Survey and Framework. ACM Comput. Surv. 55, 8, Article 170 (August 2023), 38 pages. https://doi.org/10.1145/3556536

Experiment Type: Offline, Online, User Study

Type	Description
Offline	Method: simulation of user behavior based on past interactions Task: defined by the researcher, purely algorithmic Repeatability: evaluation of an arbitrary number of experiments (e.g., algorithmic settings, models) possible at low cost Scale: large dataset, large number of users Insights: quantitative, narrow (focused on the predictive performance of algorithms)
User Study	Method: user observation in live or laboratory setting Task: defined by the researcher, carried out by the user Repeatability: expensive (recruitment of users) Scale: small cohort of users Insights: quantitative and/or qualitative (live user data, logging of user actions, eye tracking, questionnaires before/during/after task)
Online	Method: real-world user observation, online field experiment Task: self-selected by the user, carried out by the user Repeatability: expensive (requires full system and users) Scale: size of the cohort of users depending on evaluation system and user base Insights: quantitative and/or qualitative (live user data, logging of user actions, questionnaires before/during/after exposure to the system)



Eva Zangerle and Christine Bauer. 2022. Evaluating Recommender Systems: Survey and Framework. ACM Comput. Surv. 55, 8, Article 170 (August 2023), 38 pages. https://doi.org/10.1145/3556536

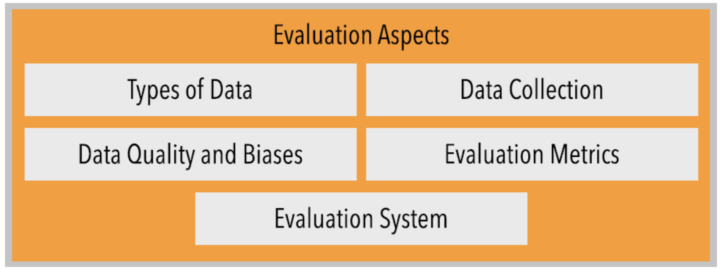
Offline Evaluation

User feedback vs user preference, the same?

- A typical experiment
  - Uses a pre-collected dataset that contains users' **explicit feedback on items** (e.g., ratings of items) or **implicit feedback on items** (e.g., the items purchased, viewed, or consumed).
  - User behavior is **mimicked and simulated** based on this historical data
  - Parts of the rating information are masked from the user-item matrix, the recommender algorithms are evaluated by their **ability to predict the missing information**
- Adoption
  - More than **92%** of the 117 RS papers published at AAAI and IJCAI in 2018 and 2019 relied exclusively on **offline experiments**. At ACM RecSys 2018 and 2019, three of four papers only used offline evaluations.
- A key issue: which values are to be masked for prediction
  - Temporal aspects of data can be critical in the design of such an evaluation

Eva Zangerle and Christine Bauer. 2022. Evaluating Recommender Systems: Survey and Framework. ACM Comput. Surv. 55, 8, Article 170 (August 2023), 38 pages. https://doi.org/10.1145/3556536

Evaluation Aspects



- Types of data
  - Implicit and explicit rating data;
  - User, item information (or side information), useful for cold-start setting
  - Qualitative and Quantitative Data
  - Natural and Synthetic Data
- Data collection
- Data quality and biases
  - Biases may occur in the distributions of users, items, or ratings that are selected to be part of the evaluation dataset
- Evaluation system
  - An interface for the evaluation, typically not applicable for offline evaluation

Table 4. Widely Used Datasets for Evaluating RS

Dataset	Domain	Size
MovieLens20M[9] [97]	Movie ratings	20,000,263 ratings; range [0.5,5]
MovieLens1M[10] [97]	Movie ratings	1,000,209 ratings; range [1,5]
BookCrossing[11] [231]	Book ratings	1,157,112 ratings; range [1,10]
Yelp[12]	Business ratings	8,021,122 ratings; range [0,5]
MovieTweetings[13] [64]	Movie ratings	871,272 ratings; range [0,10]

Eva Zangerle and Christine Bauer. 2022. Evaluating Recommender Systems: Survey and Framework. ACM Comput. Surv. 55, 8, Article 170 (August 2023), 38 pages. https://doi.org/10.1145/3556536

Evaluation Aspects

Types of Data | Data Collection

Data Quality and Biases | Evaluation Metrics

Evaluation System

Evaluation Metrics

Category	Metrics
Prediction accuracy	Mean absolute error (MAE) (Root) Mean squared error ((R)MSE)
Usage prediction	Recall, precision, F-score Receiver operating characteristic curve (ROC) Area under ROC curve (AUC)
Ranking	Normalized discounted cumulative gain (NDCG) Mean reciprocal rank (MRR)

Novelty	Item novelty Global long-tail novelty
Diversity	intra-list similarity (ILS)
Coverage	Item coverage User space coverage Gini index
Serendipity	Unexpectedness Serendipity
Fairness across users	Value unfairness Absolute unfairness Over/underestimation of fairness
Fairness across items	Pairwise fairness Disparate treatment ratio (DTR) Equal expected exposure Equity of amortized attention Disparate impact ratio (DIR) Viable-Λ test
Business-oriented	Click-through rate (CTR) Adoption and conversion rate Sales and revenue

Recall, Precision, Hit Rate, NDCG are more widely adopted in offline evaluation in academic research

Eva Zangerle and Christine Bauer. 2022. Evaluating Recommender Systems: Survey and Framework. ACM Comput. Surv. 55, 8, Article 170 (August 2023), 38 pages. https://doi.org/10.1145/3556536

Industrial Recommender System Evaluation

- E-commerce recommender system
  - Gross merchandise volume (GMV)
  - Click-through rate (CTR)
  - Conversion rate (CVR)
- Advertising-aware recommender system
  - Viewing, clicking, conversion,
  - Click-through rate (CTR)
  - Conversion rate (CVR)
- Online content recommender system: news, music, video
  - Proportion of total time spent watching, Video View, etc.

Outline

- Recommender system basics
  - Recommender system evaluation
  - Commonly used metrics in academic research and practice
- Challenges in computing the offline metrics
  - Data partition schemes in RecSys experiments using offline datasets
  - Data leakage due to not maintaining global timeline
  - The impact on understanding the RecSys research problem
- Criticism on RecSys from evaluation perspective
  - The counter-intuitive observations
  - The common pitfalls in evaluating RecSys
- More practical evaluations
  - The meaning of fair comparison
  - The observation of global timeline

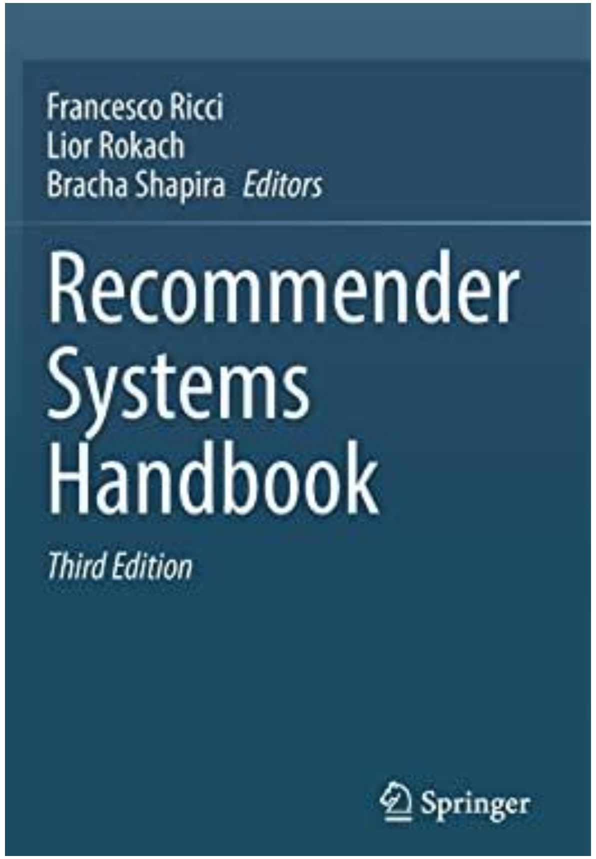
RecSys evaluation, in academic and in practice?

Data	Static dataset	Stream data
Evaluation	Train/test split	A/B testing
Metric	HitRate, NDCG…	CTR, CVR, GMV…
Model	A single model	Mixture of models?

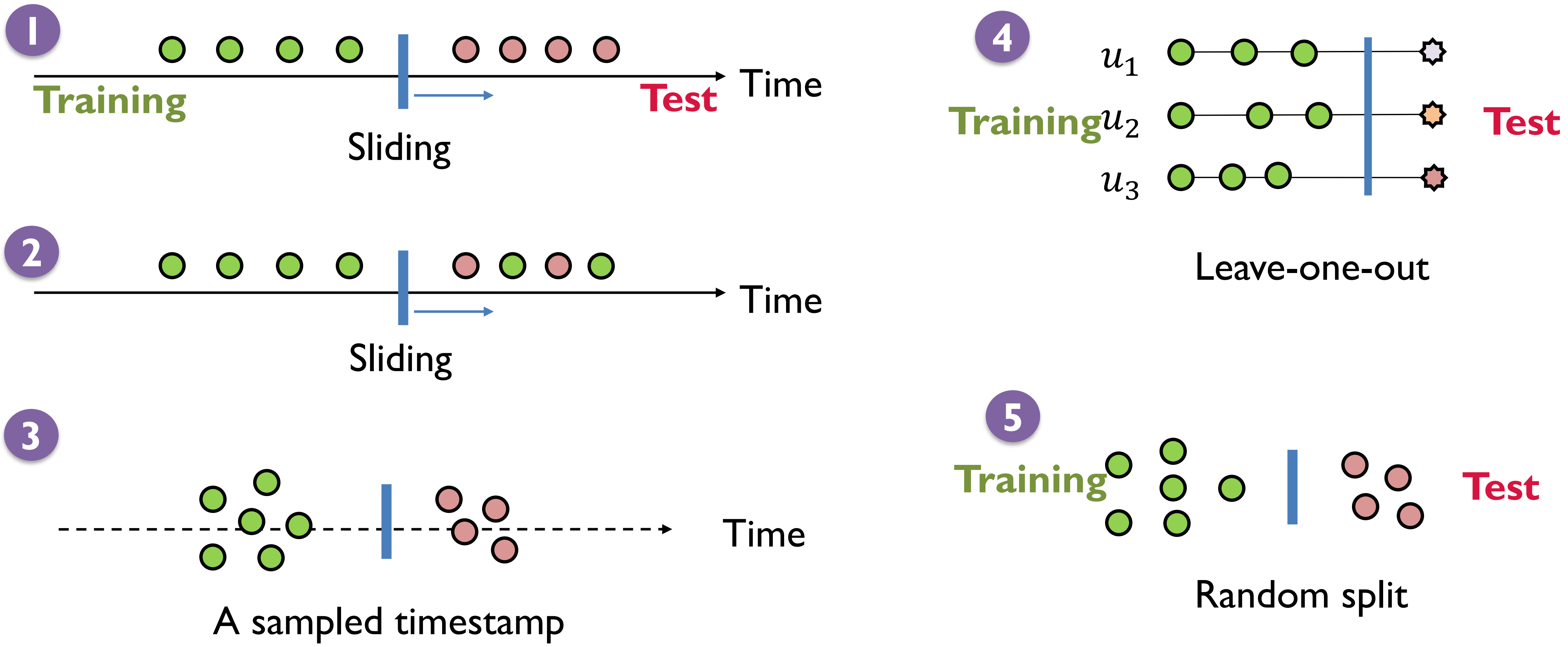
Evaluation | Train/test split | A/B testing

- “The goal of the offline experiments is to filter out inappropriate approaches, leaving a relatively small set of candidate algorithms to be tested” online

- “It is necessary to **simulate the online process** where the system makes predictions or recommendations”



The 5 settings in offline evaluation



Case study: what train/split?

- Collection: 88 papers in RecSys conferences (2020 – 2022)

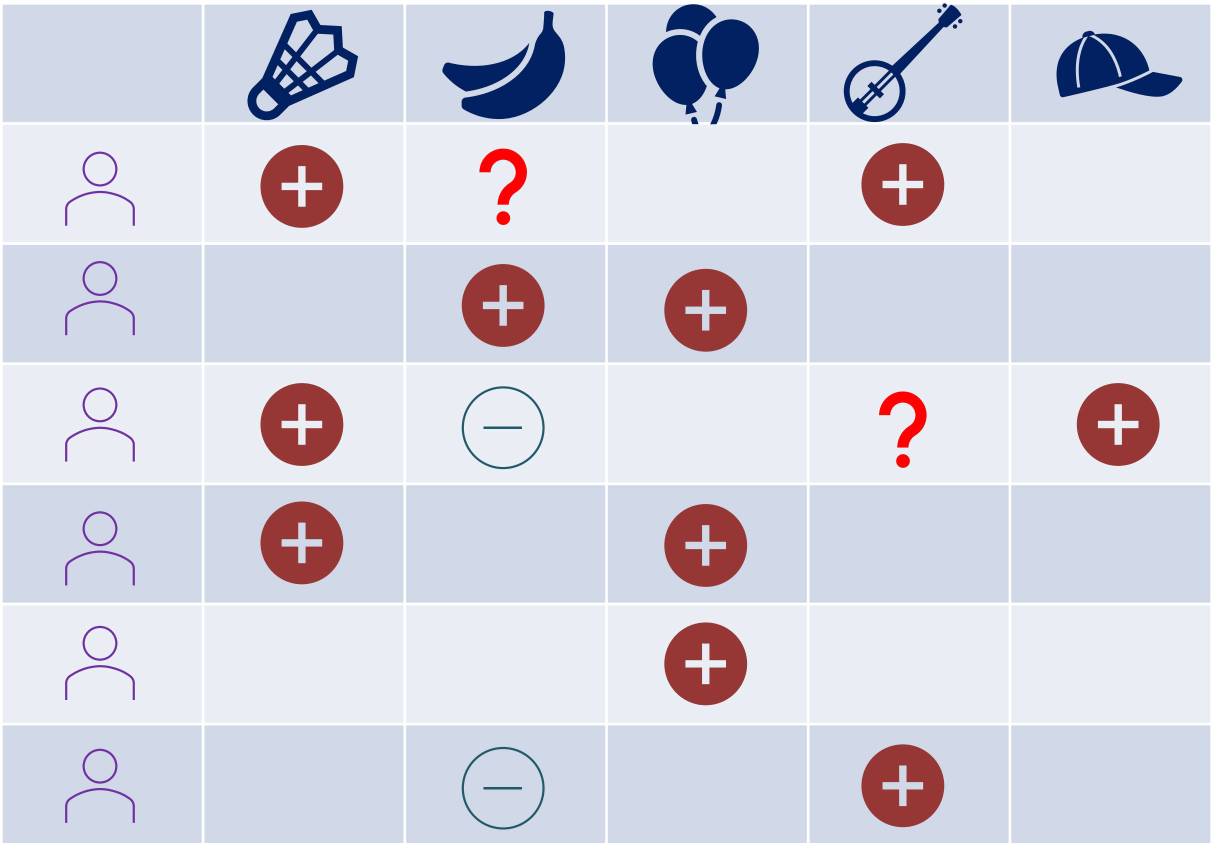
No. papers	Percentage	Train/test split	Global timeline?
30	**34%**	Random split	No
22	**25%**	Leave-one-out	No
17	19.5%	Single time point	Partially
15	17%	Simulation-based online	Yes
4	4.5%	Sliding window	Yes

Bandits and reinforcement learning for recommendation.
Incremental learning or session-based learning.

RecSys in academic research: problem abstraction

One problem definition for many RecSys tasks

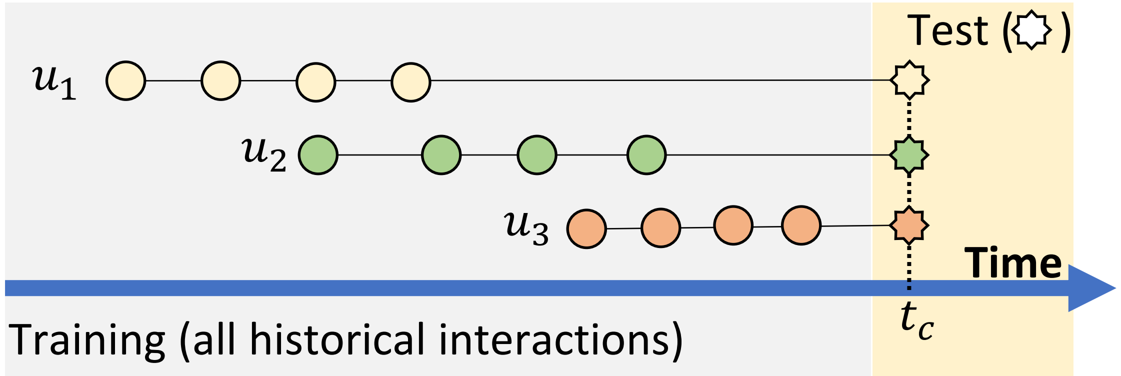
Global timeline not observed



Recommendation in practice

- Users get recommendations when visiting a site or app, at current time t_c

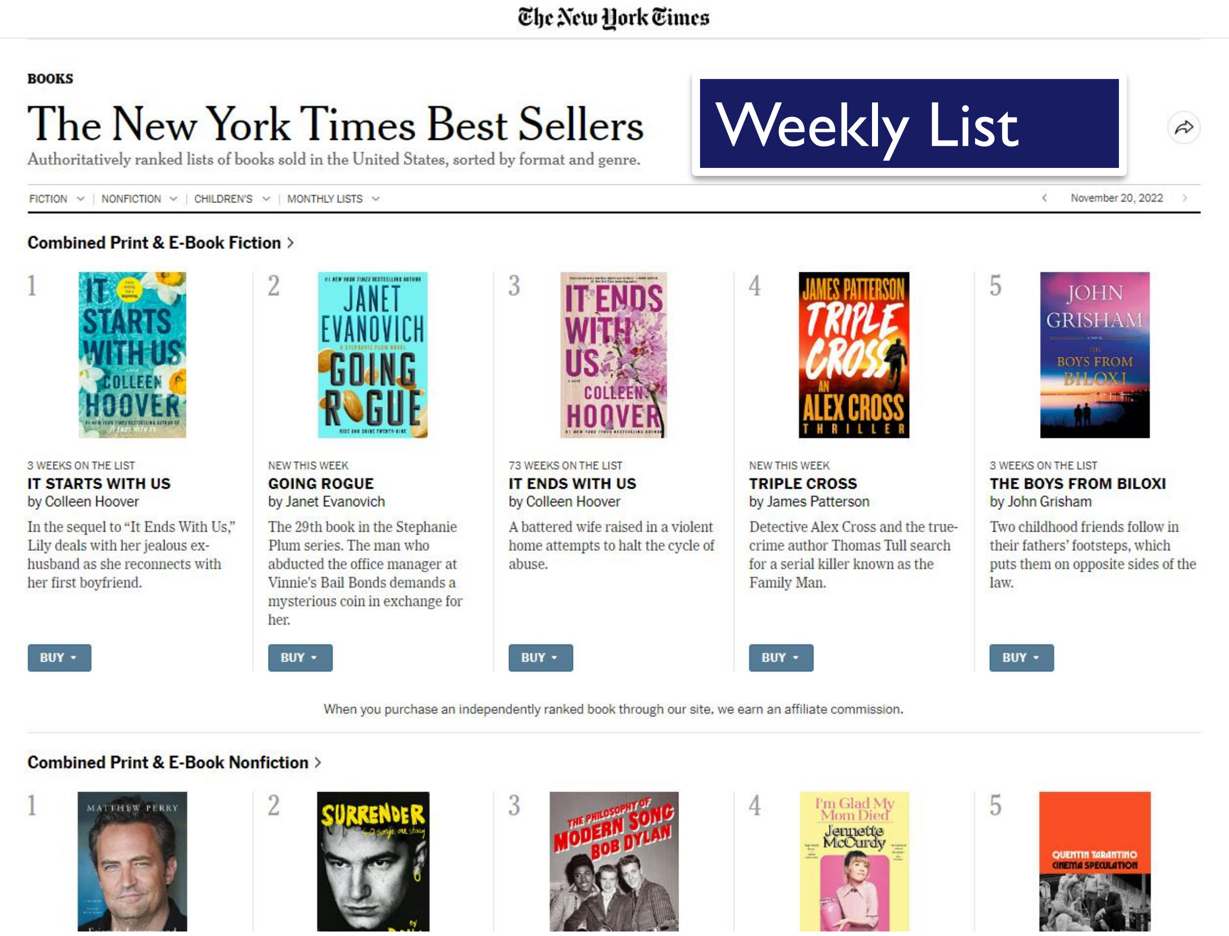
- All historical interactions before t_c can be used as training data

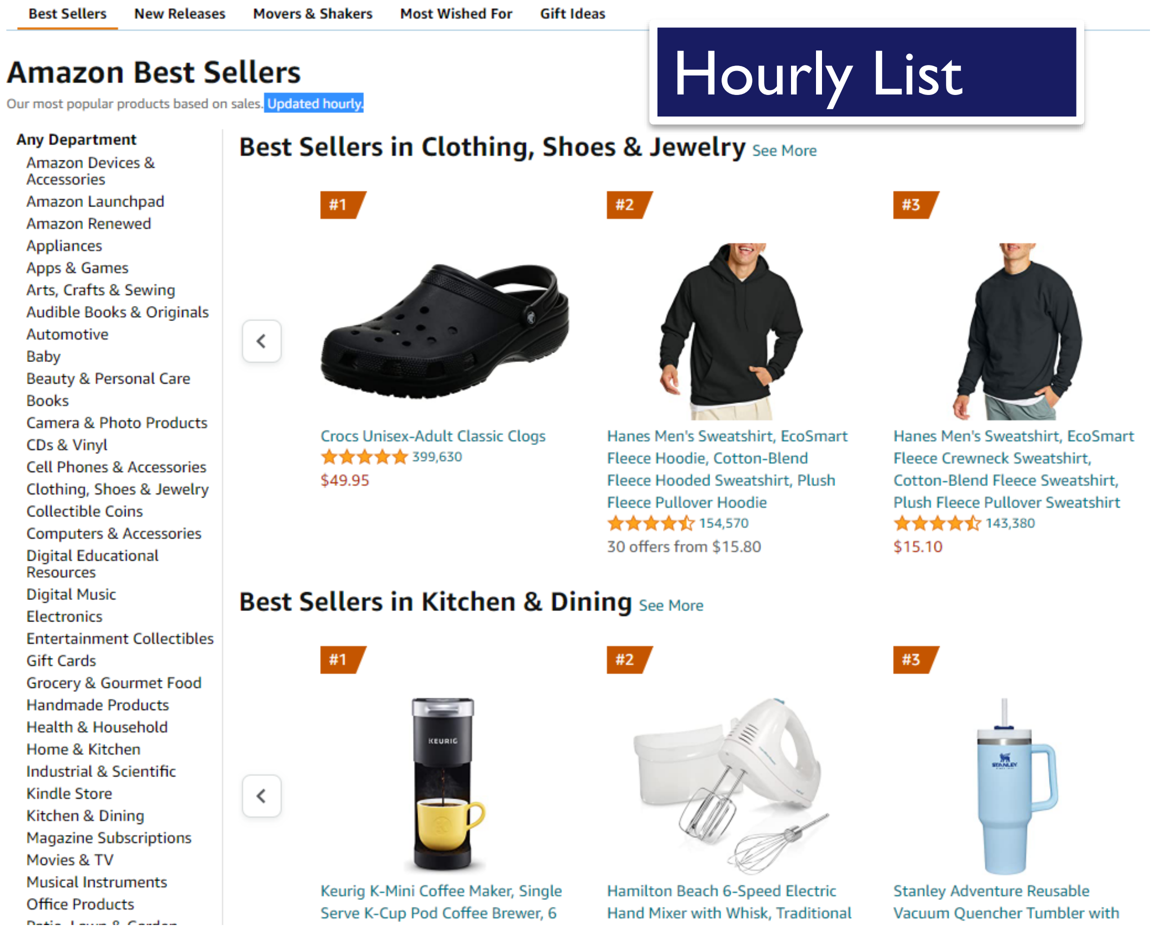


- Learning from ***past interactions***

- To ***predict*** users' preferred items ***in (near) future***

The simplest baseline: Popularity





Popularity in practice vs popularity in academic research

- Popularity in practice
  - Ranking is dynamic, updated along time
  - Ranking is based on interactions within a short time period, e.g., a week

- Popularity in academic research
  - Ranking is static, without scheduled update
  - Ranking is derived from the **entire training set**

Why is popularity defined in this way?

“**fair comparison**”

- Most **machine/deep learning** models in academic research
  - Ranking is static
  - Ranking is derived from the entire training set

Ignorance of global timeline: Data Leakage

- Recommenders access user-item interactions that "would happen" after the test time point
- Recommenders may recommend "future items"
- Recommendation accuracies may not mean much


ACM DIGITAL LIBRARY

ACM Transactions on Information Systems

RESEARCH-ARTICLE

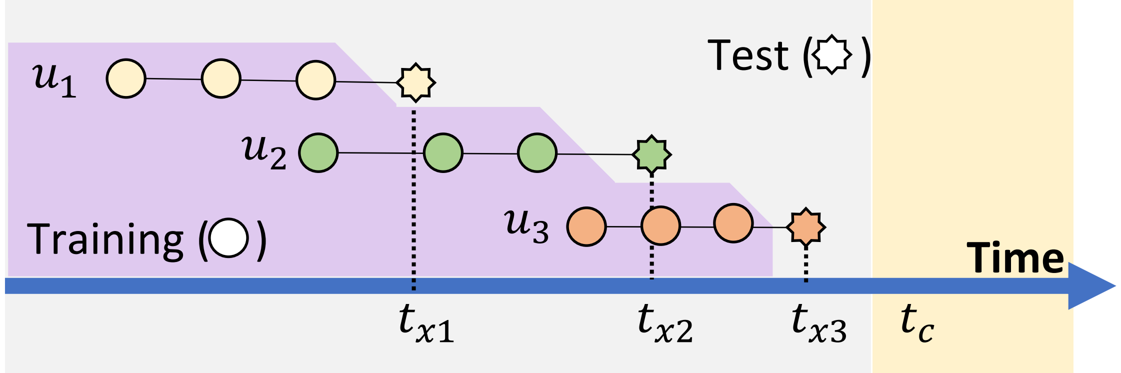
A Critical Study on Data Leakage in Recommender System Offline Evaluation

Authors: Yitong Ji, Aixin Sun, Jie Zhang, Chenliang Li Authors Info & Claims

ACM Transactions on Information Systems, Volume 41, Issue 3 • Article No.: 75, pp 1–27 • https://doi.org/10.1145/3569930

Published: 07 February 2023 Publication History Check for updates

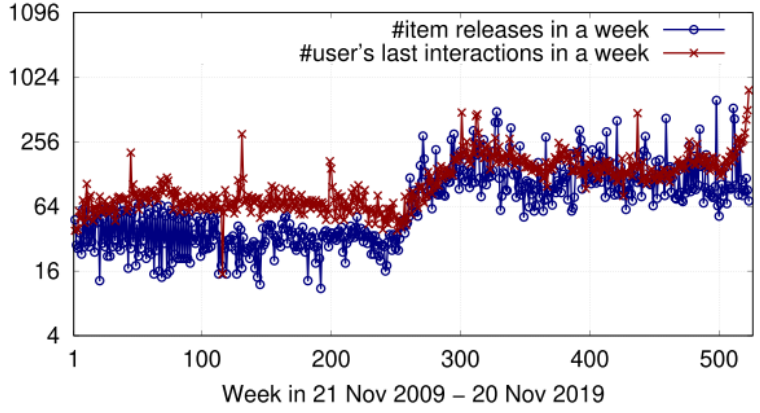
An illustration: Leave-last-one-out



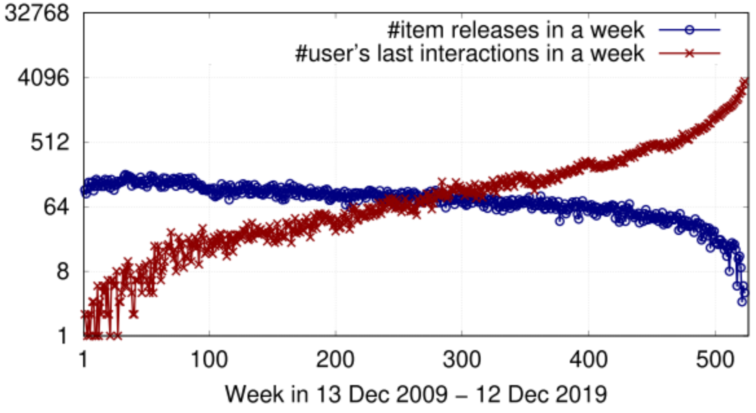
Applicable to Popularity and ML/DL-based models

Global timeline vs Local timeline

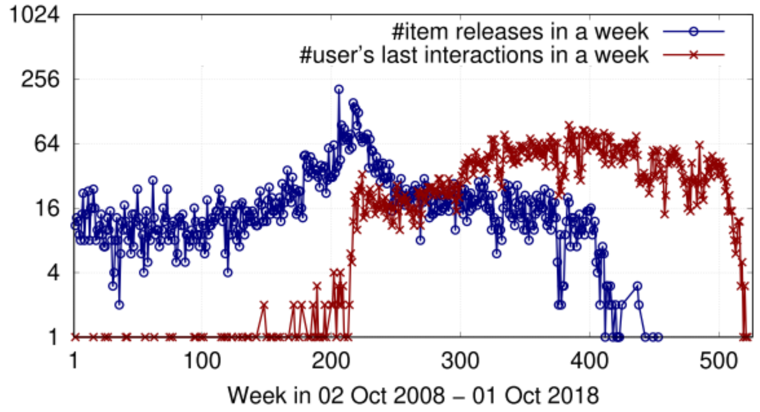
- Number of item first interactions in each week
- Number of user last interactions in each week
- On all 4 datasets for 10 years duration



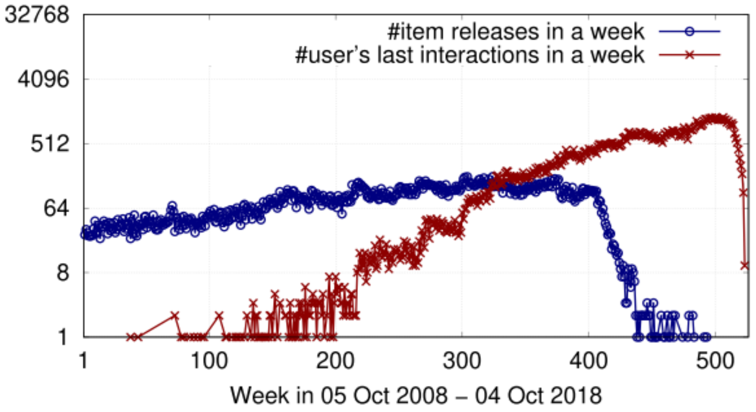
(a) MovieLens-25M



(b) Yelp

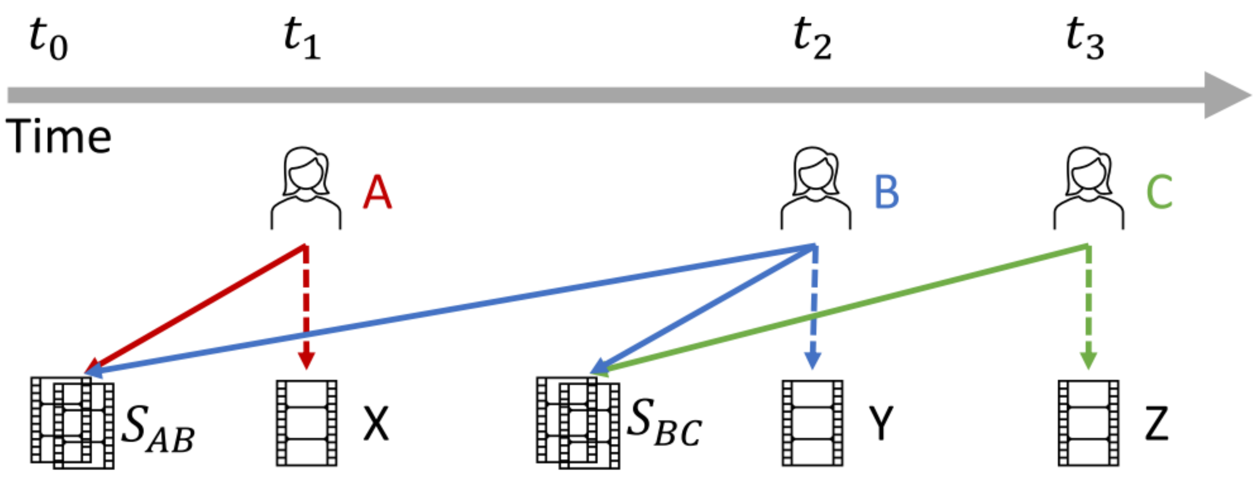


(c) Amazon-music



(d) Amazon-electronic

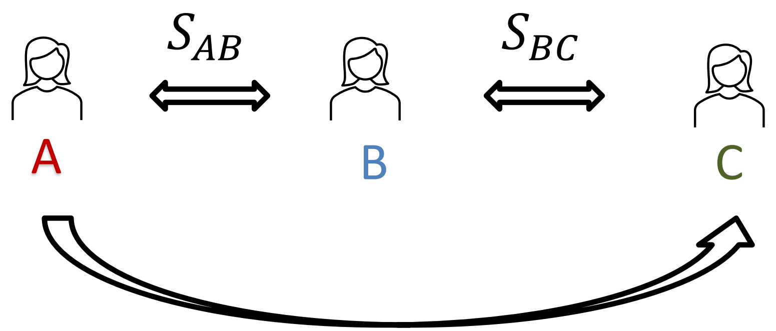
Data leakage in offline evaluation of recommender system

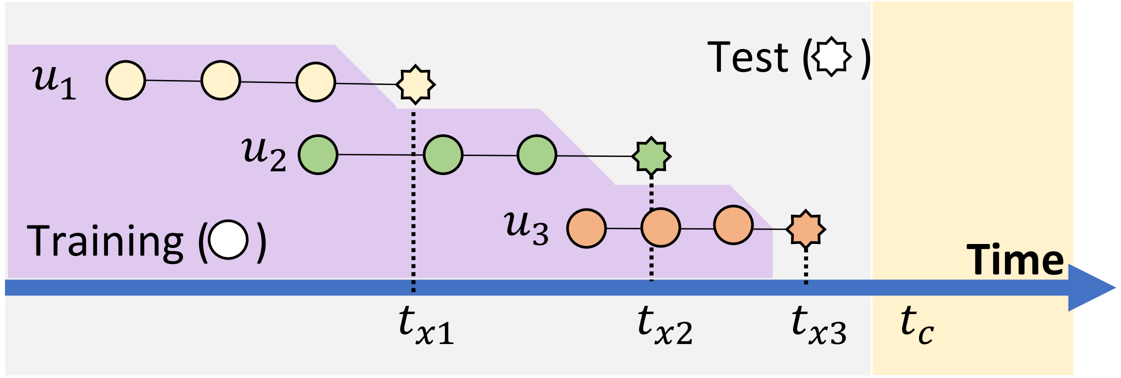


(a) User-item interaction along global timeline.

S_{AB}: items rated by both users A and B
S_{BC}: items rated by both users B and C

X: test instance of user A
Y: test instance of user B
Z: test instance of user C





All interactions by user C happened after the test instance of A

Experiments: the impact of data leakage

Dataset	Time span selected	Data Filtering	#User	#Item	#Rating	Sparsity
MovieLens-25M	21 Nov 2009 to 20 Nov 2019	No filtering	62,202	56,774	9,808,925	$2.78e-3$
Yelp	13 Dec 2009 to 12 Dec 2019	10-core	116,655	61,027	3,127,215	$4.39e-4$
Amazon-music	02 Oct 2008 to 01 Oct 2018	5-core	15,839	11,071	162,880	$9.29e-4$
Amazon-electronic	05 Oct 2008 to 04 Oct 2018	10-core	141,633	49,325	2,365,483	$3.38e-4$

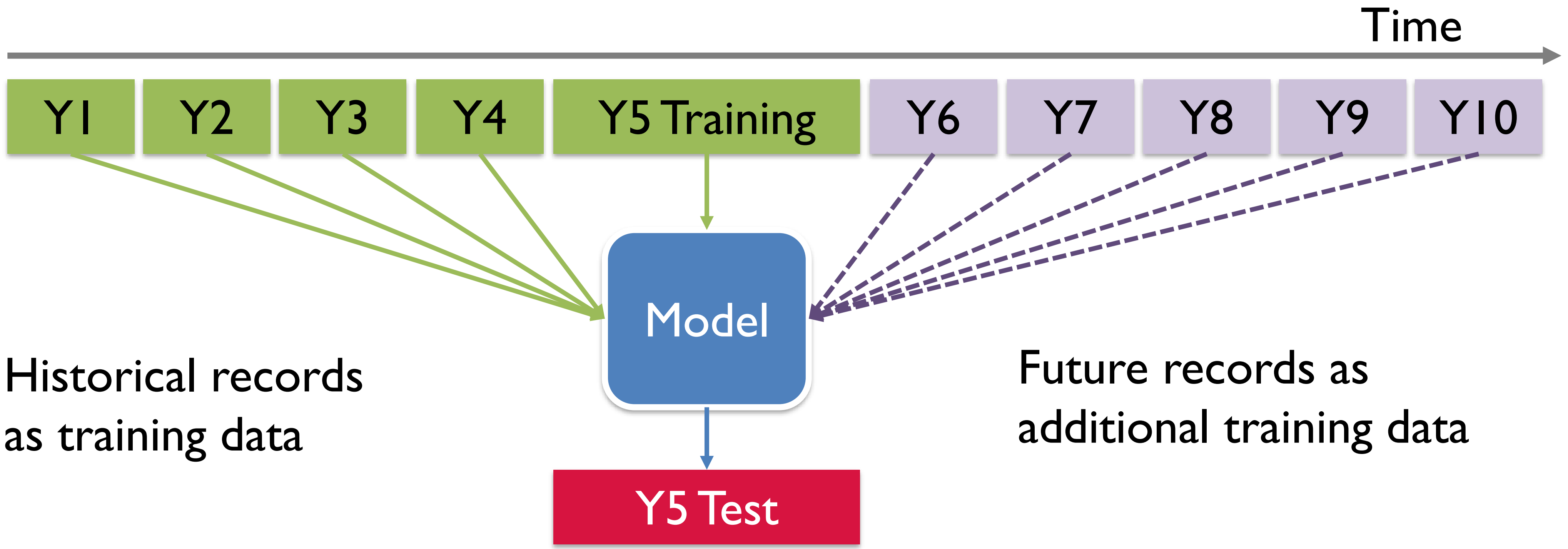
- Data partition: Leave-one-out splitting
- Baselines: BPR, NeuMF, LightGCN, SASRec
- Evaluation metrics: HR@20, NDCG@20

Recommendation List

Recommendation Accuracy

Experiment: to simulate different severity of data leakage

- Test set: test instances that happened in Year 5 (example test year)
- Training set: (Instances before Y5) + (training instances in Y5) + (x year of future instances), $x \in [0,5]$



Impact of data leakage on recommendation list

- **Future items**: the items are exclusively available only after the specific time point of a given test instance.

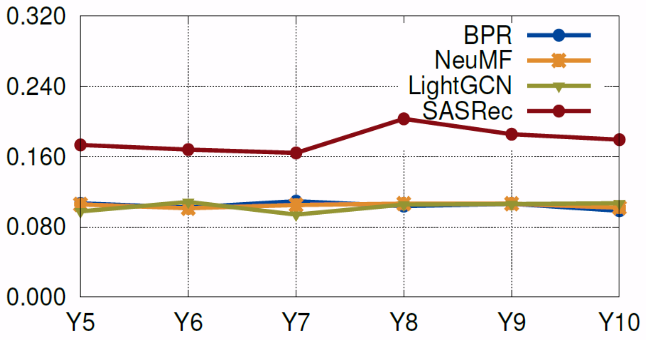
- All models recommend "future items" → **invalid recommendation**

Model	Dataset / Test year	MovieLens-25M Y5	MovieLens-25M Y7	Yelp Y5	Yelp Y7	Amazon-music Y5	Amazon-music Y7	Amazon-electronic Y5	Amazon-electronic Y7
BPR	Y5	0	–	0	–	0	–	0	–
	Y6	0	–	421	–	615	–	79	–
	Y7	22	0	829	0	970	0	363	0
	Y8	7	11	2,365	504	1,101	651	263	200
	Y9	6	88	5,048	287	1,304	1,103	499	1,224
	Y10	4	81	1,851	1,598	1,197	1,155	200	583
NeuMF	Y5	0	–	0	–	0	–	0	–
	Y6	3	–	602	–	910	–	28	–
	Y7	7	0	1,631	0	1,501	0	1,303	0
	Y8	27	31	3,260	130	1,733	878	549	0
	Y9	22	6	3,542	1,177	1,491	1,276	729	216
	Y10	15	1	5,205	1,791	1,577	1,573	2,655	326
LightGCN	Y5	0	–	0	–	0	–	0	–
	Y6	11	–	369	–	626	–	37	–
	Y7	32	0	739	0	1,050	0	148	0
	Y8	116	189	1,070	569	998	632	367	220
	Y9	22	26	1,257	979	1,036	893	262	430
	Y10	15	58	1,103	1,360	1,152	1,029	260	470
SASRec	Y5	0	–	0	–	0	–	0	–
	Y6	315	–	967	–	906	–	216	–
	Y7	442	0	3,074	0	1,548	0	625	0
	Y8	144	489	2,228	2,666	1,814	1,341	487	1388
	Y9	342	403	3,162	2,893	1,982	1,376	20	3,209
	Y10	993	386	1,741	3,014	1,980	1,662	12	2,479

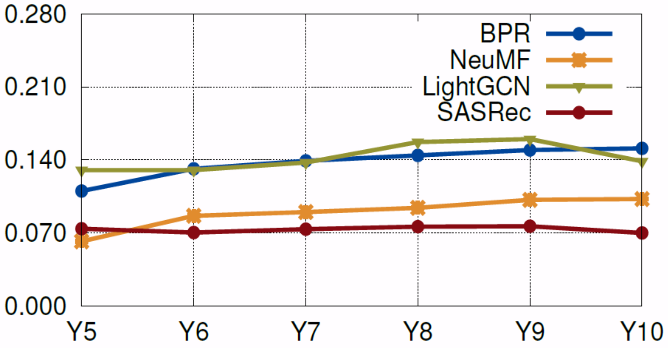
Impact of data leakage on recommendation accuracy

- The impact on recommendation accuracy can vary, and it is **not predictable**.

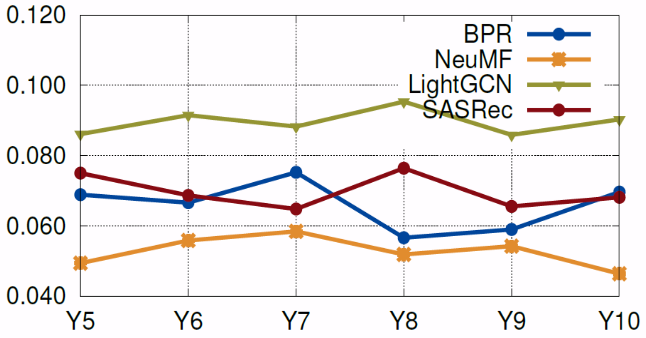
- The **relative performance ordering** of the evaluated models does not exhibit consistent patterns.



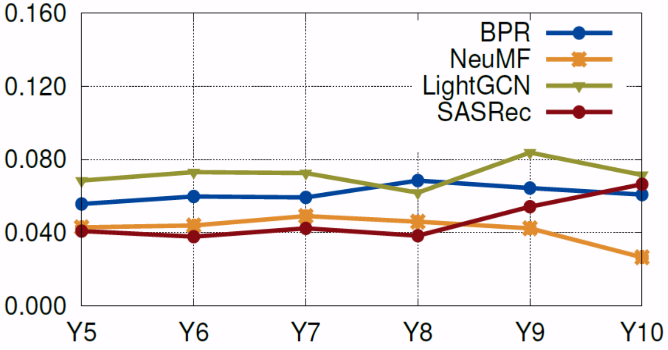
(A) HR@20
MovieLens-25M



(E) HR@20
Amazon-music



(C) HR@20
Yelp

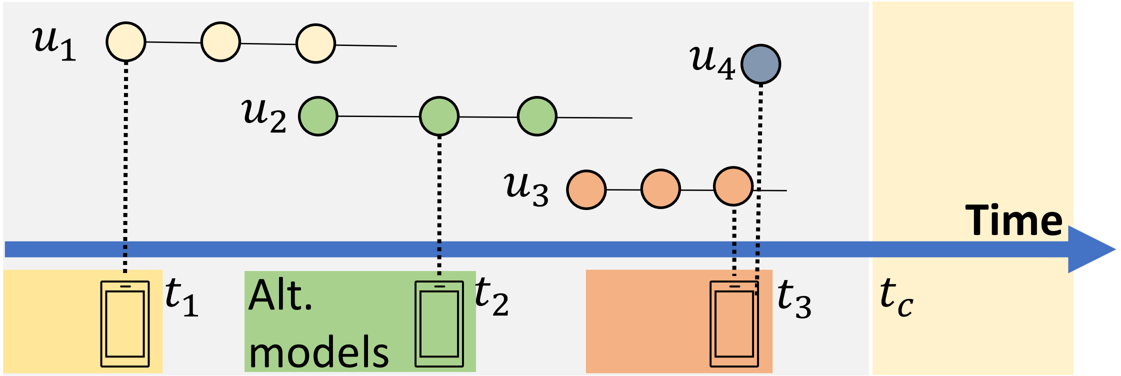


(G) HR@20
Amazon-electronic

Ignorance of global timeline: Simplified User Preference Learning

All users u_1 to u_4 purchased the same phone, but at different time points

- User u_1 purchased iPhone X on its first day of release
- Users u_3 and u_4 purchased iPhone X when the next model was released.
- User u_2 purchased iPhone X some day in between.



Are all decision-makings the same?

What reflects user preference?
(a) decision making process,
(b) result of decision?

Re-visiting collaborative filtering

Communications of the ACM Dec 1992 v35 n12 p61(10) Page 1

Using collaborative filtering to weave an information Tapestry.

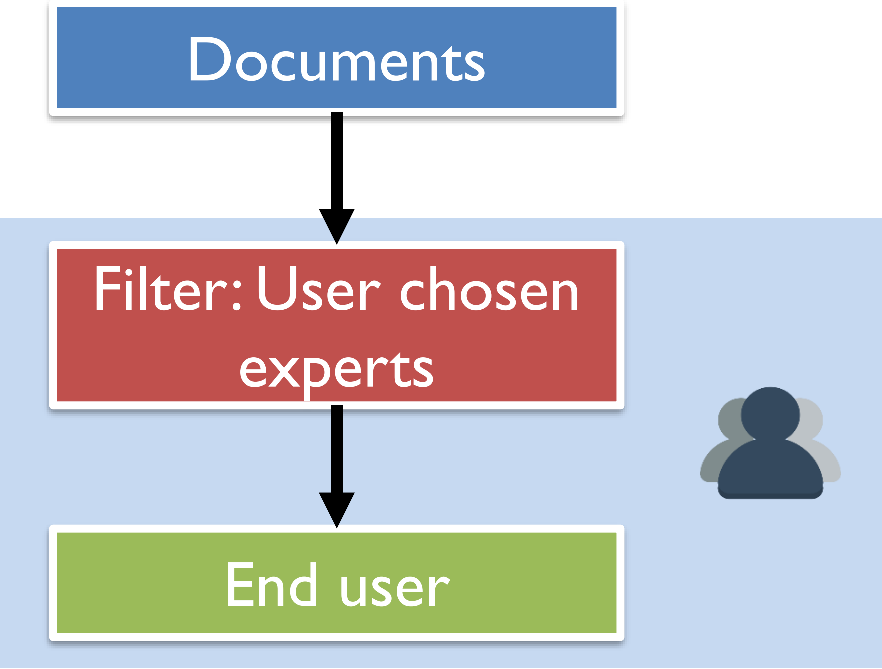
by David Goldberg, David Nichols, Brian M. Oki and Douglas Terry

The Tapestry experimental mail system developed at the Xerox Palo Alto Research Center is predicated on the belief that information filtering can be more effective when humans are involved in the filtering process. Tapestry was designed to support both content-based filtering and collaborative filtering, which entails people collaborating to help each other perform filtering by recording their reactions to documents they read. The reactions are called annotations; they can be accessed by other people's filters. Tapestry is intended to handle any incoming stream of electronic documents and serves both as a mail filter and repository; its components are the indexer, document store, annotation store, filterer, little box, remailer, appraiser and reader/browser. Tapestry's client/server architecture, its various components, and the Tapestry query language are described.

- A user wants to read interesting but not all documents from a newsgroup.
  - She knows that some users read all of these documents and mark the interesting ones.
  - She then can simply choose to read only the documents that are **marked interesting by these users**.
- Tapestry allows a user to filter documents by "**users with similar preference**"

https://doi.org/10.1145/138859.138867

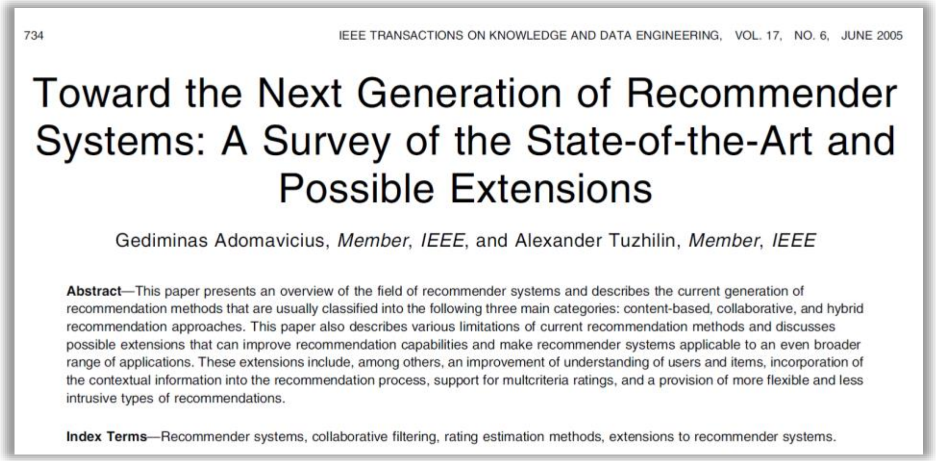
Collaborative filtering: 1992



- User does not want to access all documents
- User trusts "recommendations" by self-defined "experts"
- Recommendation → **information filter**
  - Twitter
  - Facebook
  - LinkedIn

A **hypothetical** extension:
if user u_1 follows u_2, then u_1 prefers u_2's **decision making** in judging interesting documents, given the **context at that time**, e.g., when a document is received in the newsgroup

Recommender System – 2005


734 IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 17, NO. 6, JUNE 2005

Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions

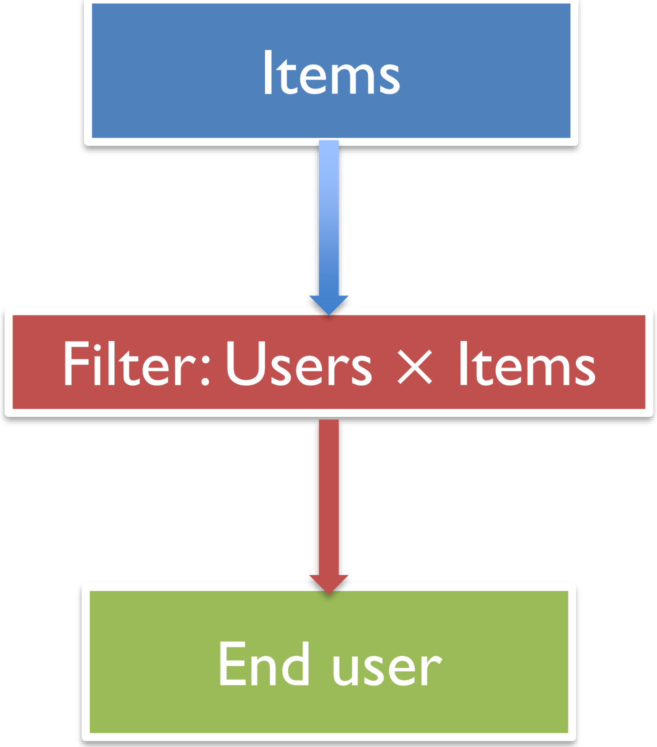
Gediminas Adomavicius, *Member, IEEE*, and Alexander Tuzhilin, *Member, IEEE*

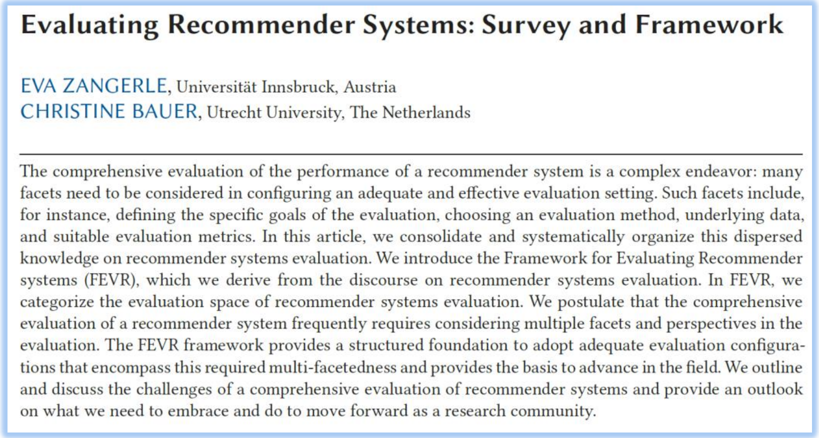
Abstract—This paper presents an overview of the field of recommender systems and describes the current generation of recommendation methods that are usually classified into the following three main categories: content-based, collaborative, and hybrid recommendation approaches. This paper also describes various limitations of current recommendation methods and discusses possible extensions that can improve recommendation capabilities and make recommender systems applicable to an even broader range of applications. These extensions include, among others, an improvement of understanding of users and items, incorporation of the contextual information into the recommendation process, support for multcriteria ratings, and a provision of more flexible and less intrusive types of recommendations.

Index Terms—Recommender systems, collaborative filtering, rating estimation methods, extensions to recommender systems.

Collaborative filtering

- The most dominant approach for computing recommendations
- Based on the collective behavior of a system's users: user-item interaction matrix
- **Assumption: users who had similar preferences in the past will also have similar preferences in the future.**

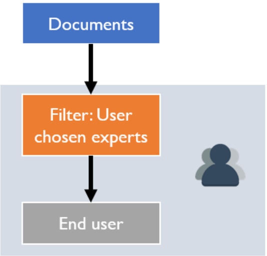



Evaluating Recommender Systems: Survey and Framework

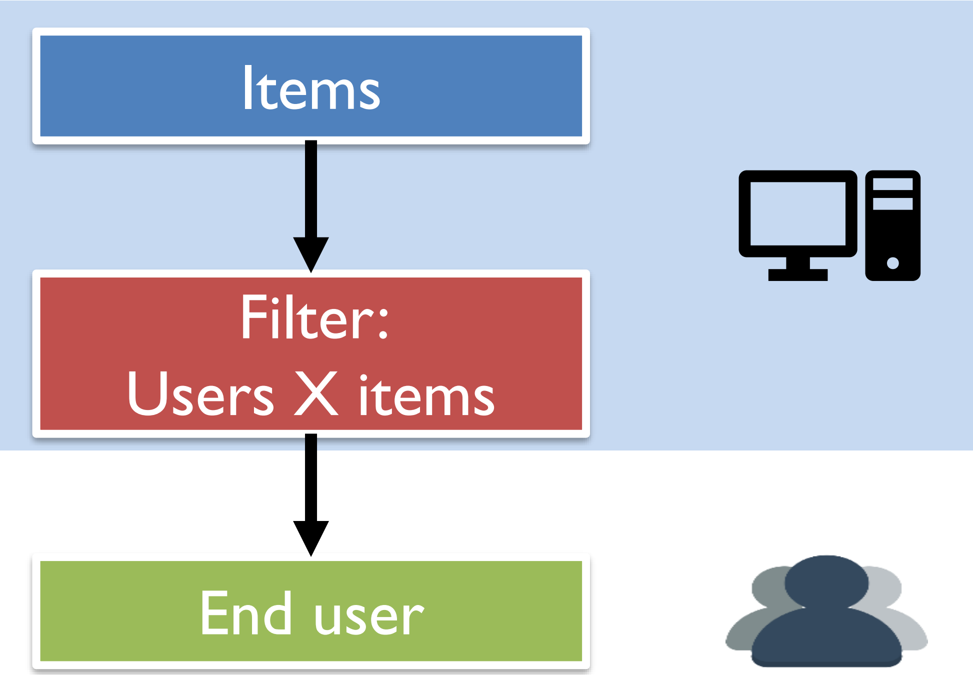
EVA ZANGERLE, Universität Innsbruck, Austria
CHRISTINE BAUER, Utrecht University, The Netherlands

The comprehensive evaluation of the performance of a recommender system is a complex endeavor: many facets need to be considered in configuring an adequate and effective evaluation setting. Such facets include, for instance, defining the specific goals of the evaluation, choosing an evaluation method, underlying data, and suitable evaluation metrics. In this article, we consolidate and systematically organize this dispersed knowledge on recommender systems evaluation. We introduce the Framework for Evaluating Recommender systems (FEVR), which we derive from the discourse on recommender systems evaluation. In FEVR, we categorize the evaluation space of recommender systems evaluation. We postulate that the comprehensive evaluation of a recommender system frequently requires considering multiple facets and perspectives in the evaluation. The FEVR framework provides a structured foundation to adopt adequate evaluation configurations that encompass this required multi-facetedness and provides the basis to advance in the field. We outline and discuss the challenges of a comprehensive evaluation of recommender systems and provide an outlook on what we need to embrace and do to move forward as a research community.

User information needs: Defined by other "similar" users



Collaborative filtering: the current understanding



- A user u would prefer the items that are chosen by other users who share similar preferences with u.
- Preference similarity between users is reflected by **similar user-item interactions** in the past.
- If users u_1 and u_2 both purchased the same mobile phone, then we would consider that u_1 and u_2 share similar preference, at least on this particular item.

Does purchasing the same item reflect that the two users share a similar **decision-making process**?
Do we need to consider the context changes in from time to time?

The possible context changes in decision making

- Even if two users interact with the same item,
  - If the two interactions occur at very different time points, the contexts for the two decision makings could be very different.
  - The context here is reflected by **the candidate items and their properties** (e.g., their popularity ranking) at the "decision making" time

- There are many context changes
  - User side: moved to a new city, changed office, salary increase, graduated…..
  - System side: Item ranking changes, competitive alternatives … (we only consider the changes that can be observed through the data)

- More reasonable to assume that if two interactions occur within a short time period, the context change at system side is not significant.

Outline

- Recommender system basics
  - Recommender system evaluation
  - Commonly used metrics in academic research and practice
- Challenges in computing the offline metrics
  - Data partition schemes in RecSys experiments using offline datasets
  - Data leakage due to not maintaining global timeline
  - The impact on understanding the RecSys research problem
- Criticism on RecSys from evaluation perspective
  - The counter-intuitive observations
  - The common pitfalls in evaluating RecSys
- More practical evaluations
  - The meaning of fair comparison
  - The observation of global timeline

Do Loyal Users Enjoy Better Recommendations? Understanding Recommender Accuracy from a Time Perspective

Yitong Ji
Nanyang Technological University
Singapore
yitong.ji@ntu.e

Aixin Sun
Nanyang Technological University
Singapore

Jie Zhang
Nanyang Technologica
Singapore
zhangj@ntu.ed

ABSTRACT

In academic research, recommender syste
benchmark datasets, without much consi
timeline. Hence, we are unable to answe
users enjoy better recommendations than
can be defined by the time period a user
ommender system, or by the number of
user has. In this paper, we offer a compreh
mendation results along global timeline.
with five widely used models, *i.e.*, BPR, Ne
and TiSASRec, on four benchmark datas
Yelp, Amazon-music, and Amazon-electr
sults give an answer "No" to the above qu
historical interactions suffer from relativ
tions. Users who stay with the system f
enjoy better recommendations. Both findi
Interestingly, users who have recently int
with respect to the time point of the te
recommendations. The finding on recen
gardless of users' loyalty. Our study offers
understand recommender accuracy, and o
a revisit of recommender model design.
https://github.com/putatu/recommenderl

Are We Forgetting Something? Correctly Evaluate a Recommender System With an Optimal Training Window

Robin Verachtert[1,2], Lien Michiels[1,2] and Bart Goethals[1,2,3]

[1] *Froomle N.V., Belgium*
[2] *University of Antwerp, Antwerp, Belgium*
[3] *Monash University, Melbourne, Australia*

Abstract

Recommender systems are deployed in dynamic environments with constantly changing interests and availability of items, articles and products. The hyperparameter optimisation of such systems usually happens on a static dataset, extracted from a live system Although it is well known that the quality of a computed model highly depends on the quality of the data it was trained on, this is largely neglected in these optimisations. For example, when concept drift occurs in the data, the model is likely to learn patterns that are not aligned with the target prediction data. Interestingly, most scientific articles on recommender systems typically perform their evaluation on entire datasets, without considering their intrinsic quality or that of their parts. First, we show that using only the more recent parts of a dataset can drastically improve the performance of a recommendation system, and we pose that it should be a standard hyperparameter to be tuned prior to evaluation and deployment. Second, we find that comparing the performance of well-known baseline algorithms before and after optimising the training data window significantly changes the performance ranking.

Check for updates

tes and recommendation
ry

na W

tion
tiona
to re
anal
E al
tions
rman

Experience summarizes the length and intensity of the consumer's relationship with the vendor and is based on four metrics:

1. Number of days since the account creation (mean 255.03, standard deviation 278.48).
2. Number of days since the first shopping transaction (mean 24.05, standard deviation 56.54).
3. Number of purchase transactions in the past year (mean 1.99, standard deviation 2.86).
4. Value of transactions in the past year (mean 188.20 Euro, standard deviation 822.39 Euro).

trust and engagement with the vendor. Experience with the vendor showed a negative correlation with recommendation performance through both its main effect and by its interactions with other consumer-related variables.

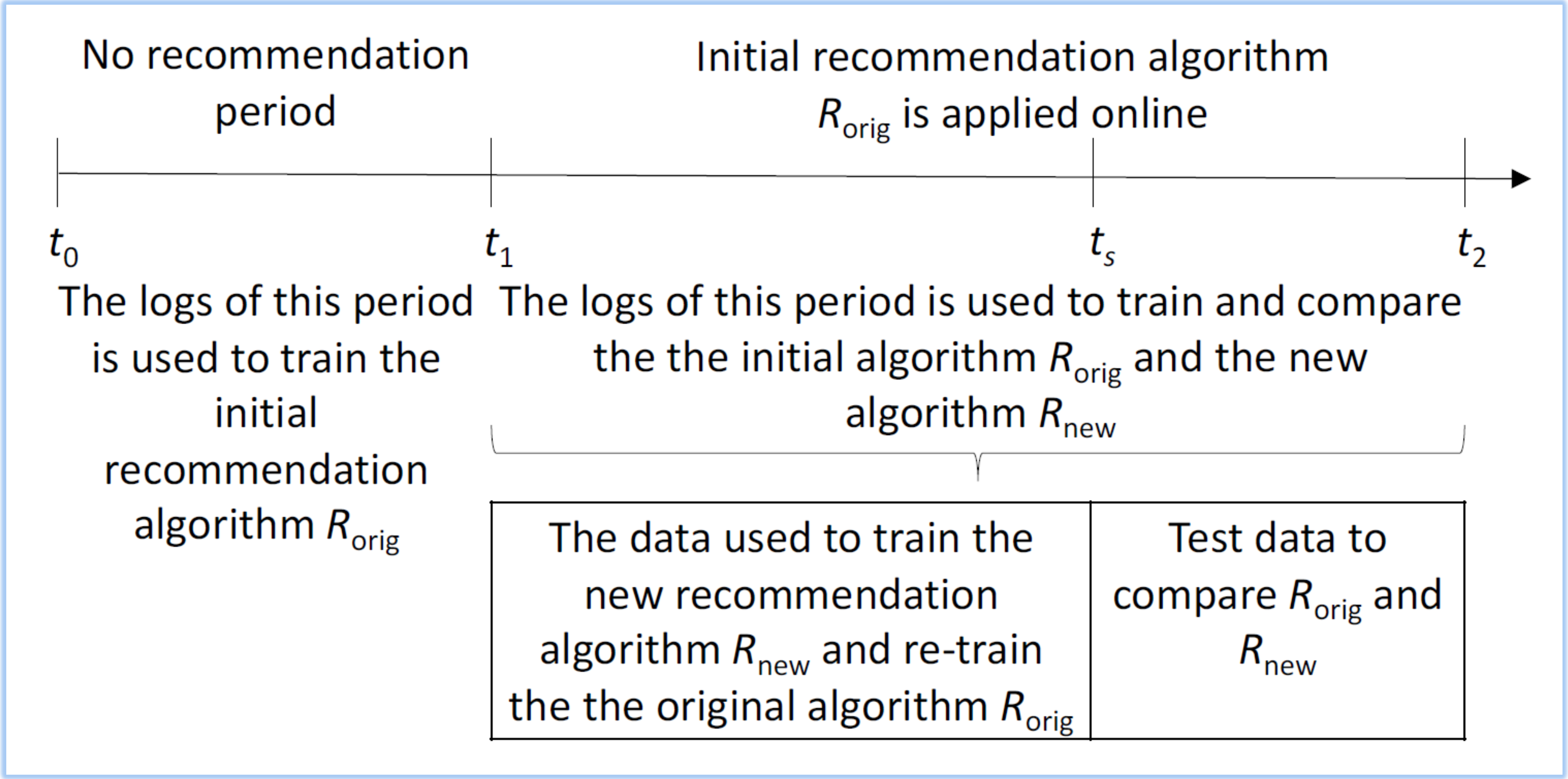
Counter-intuitive observations

- ICTIR 2022:
  - Users with **many historical interactions** suffer from relatively **poorer recommendations**.
- Electronic Markets 22:
  - **Experience** with the vendor showed a **negative correlation** with recommendation performance.
- PERSPECTIVES 2022:
  - Using only the more **recent parts of a dataset** can drastically **improve the performance** of a recommendation system

Time dimension:
Global timeline

Counter-intuitive

Common pitfalls in evaluating recommender systems



Hung-Hsuan Chen, Chu-An Chung, Hsin-Chien Huang, and Wen Tsui. 2017. Common Pitfalls in Training and Evaluating Recommender Systems. SIGKDD Explor. Newsl. 19, 1 (June 2017), 37–45. https://doi.org/10.1145/3137597.3137601

Common pitfalls in evaluating recommender systems

- Issue 1 **training data**: Clickstreams are highly influenced by the reachability of the products and the layouts of the product pages.
  - The items that occupy many spaces are more likely to be clicked and reached.
  - The trained recommender is likely to learn (1) the "layout" of the pages, and (2) the recommendation rules of the online recommender system.

- Issue 2 **test data**: If the suggested product list L_{new} recommended by the new recommendation module R_{new} is very different from the online recommendation module's list L_{org}, the online users have no chances to click on the products that appear only in L_{new} but not in L_{org}.

Hung-Hsuan Chen, Chu-An Chung, Hsin-Chien Huang, and Wen Tsui. 2017. Common Pitfalls in Training and Evaluating Recommender Systems. SIGKDD Explor. Newsl. 19, 1 (June 2017), 37–45. https://doi.org/10.1145/3137597.3137601

Common pitfalls in evaluating recommender systems

Not related to this tutorial

- **Issue 3**: Click through rates are mediocre proxy to revenues
  - User-centric measures (e.g., click through rate) vs business-centric measures (e.g., recommendation revenue).
  - Unfortunately, such a surmise was not carefully validated.

- **Issue 4**: Evaluating recommendation revenue is not straightforward
  - It is possible that the recommendation modules are served as a convenient tool for users to locate the desired items in e-commerce, but even without the recommendation module, the users can still discover these items through another means.

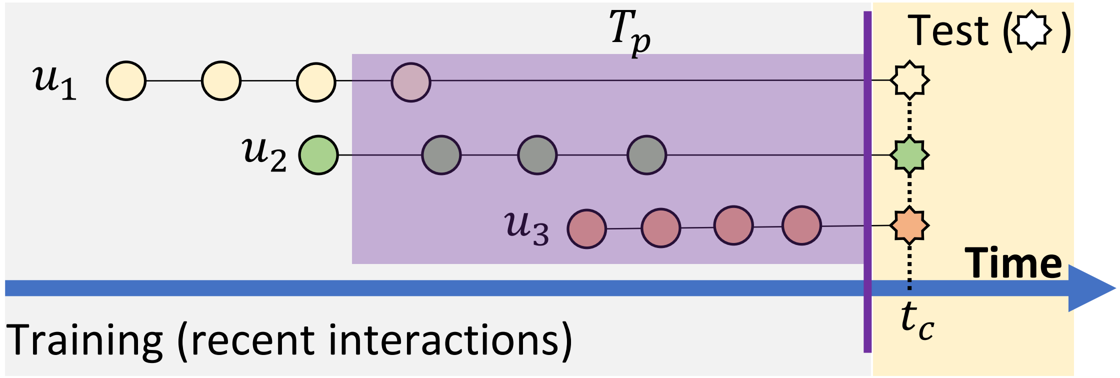
Hung-Hsuan Chen, Chu-An Chung, Hsin-Chien Huang, and Wen Tsui. 2017. Common Pitfalls in Training and Evaluating Recommender Systems. SIGKDD Explor. Newsl. 19, 1 (June 2017), 37–45. https://doi.org/10.1145/3137597.3137601

Outline

- Recommender system basics
  - Recommender system evaluation
  - Commonly used metrics in academic research and practice
- Challenges in computing the offline metrics
  - Data partition schemes in RecSys experiments using offline datasets
  - Data leakage due to not maintaining global timeline
  - The impact on understanding the RecSys research problem
- Criticism on RecSys from evaluation perspective
  - The counter-intuitive observations
  - The common pitfalls in evaluating RecSys
- More practical evaluations
  - The meaning of fair comparison
  - The observation of global timeline

RecSys evaluation is extremely challenging

- The evaluation metrics can be defined from multiple perspectives
  - Model accuracy? Business KPI?
  - Impact of website design, existing RecSys models, and many other factors



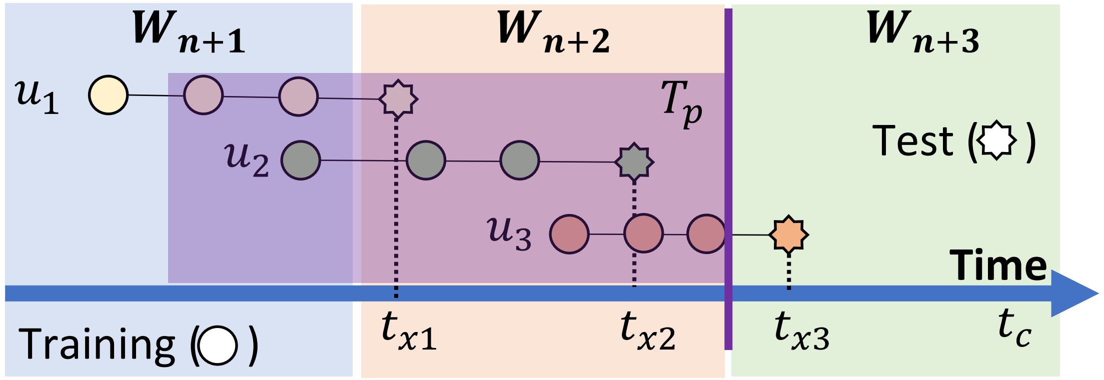
- We probably want to begin with something simple
  - A re-consideration of "**fair comparison**"
  - An evaluation protocol with **no or minimum data leakage**

Do not force "Popularity" to use all training data

Meaningful and practical evaluation

All user-item interactions (in both train and test) are arranged in chronological order.

- ➢ The entire timeline is split into time windows of size W
- ➢ One window W is tested at each time, window by window



A model may use all or subset (e.g., only recent) training data

Meaningful modeling of user preference

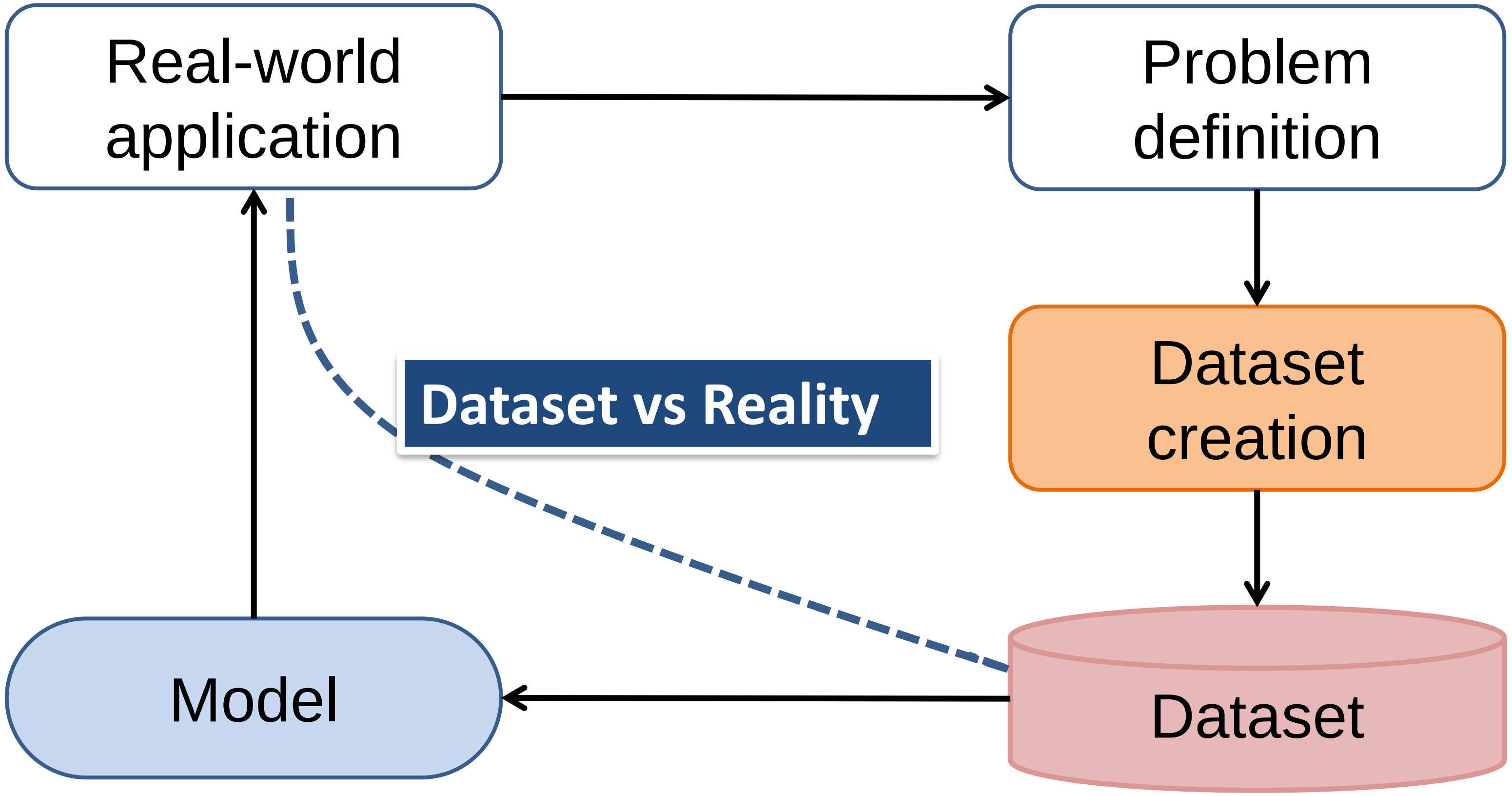
- A better understanding of user preference
  - Is decision context something worth studying?
  - What is decision context?

- Possible ways of evaluating similarity between decision contexts
  - Impressions:
    - User u_1 chooses item D with impression $\{A, B, C, D\}$, and user u_2 chooses item D with impression $\{D, E, F, G\}$, are their decision contexts the same?
  - A simplified version (assumption):
    - If two interactions happen within a very short time period, then the decision contexts are similar.

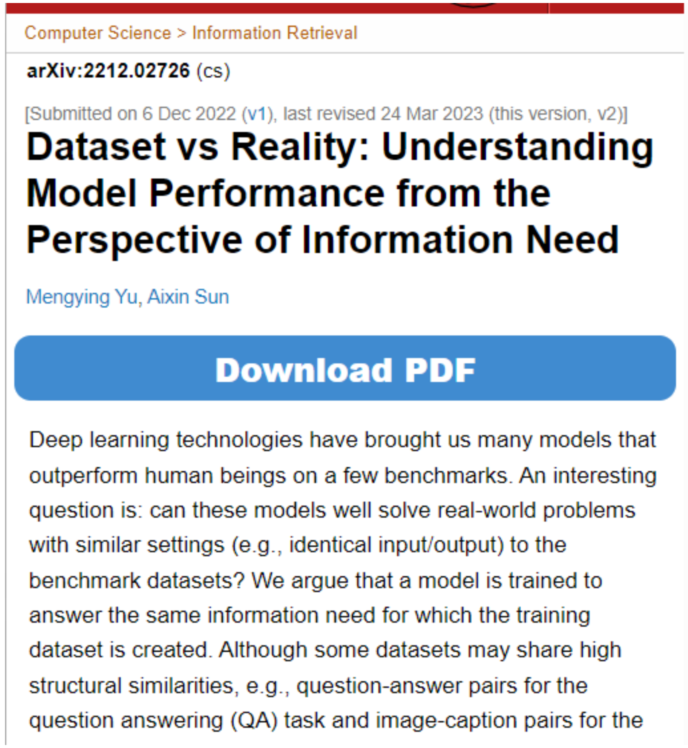
RecSys evaluation, in academic and in practice?

Data	Static dataset	Stream data
Evaluation	Train/test split	A/B testing
Metric	HitRate, NDCG…	CTR, CVR, GMV…
Model	A single model	Mixture of models?

Dataset vs Reality: An appropriate dataset for evalution



https://arxiv.org/abs/2212.02726


Computer Science > Information Retrieval

arXiv:2212.02726 (cs)

[Submitted on 6 Dec 2022 (v1), last revised 24 Mar 2023 (this version, v2)]

Dataset vs Reality: Understanding Model Performance from the Perspective of Information Need

Mengying Yu, Aixin Sun

Download PDF

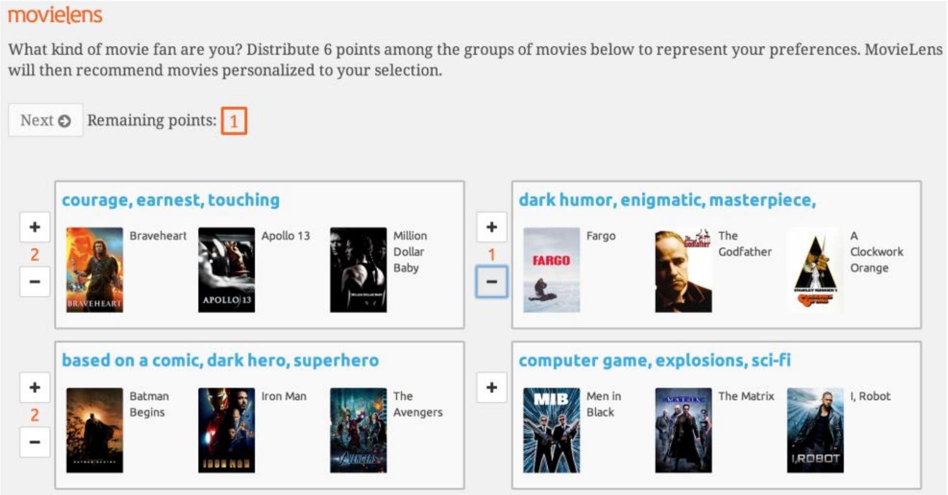
Deep learning technologies have brought us many models that outperform human beings on a few benchmarks. An interesting question is: can these models well solve real-world problems with similar settings (e.g., identical input/output) to the benchmark datasets? We argue that a model is trained to answer the same information need for which the training dataset is created. Although some datasets may share high structural similarities, e.g., question-answer pairs for the question answering (QA) task and image-caption pairs for the

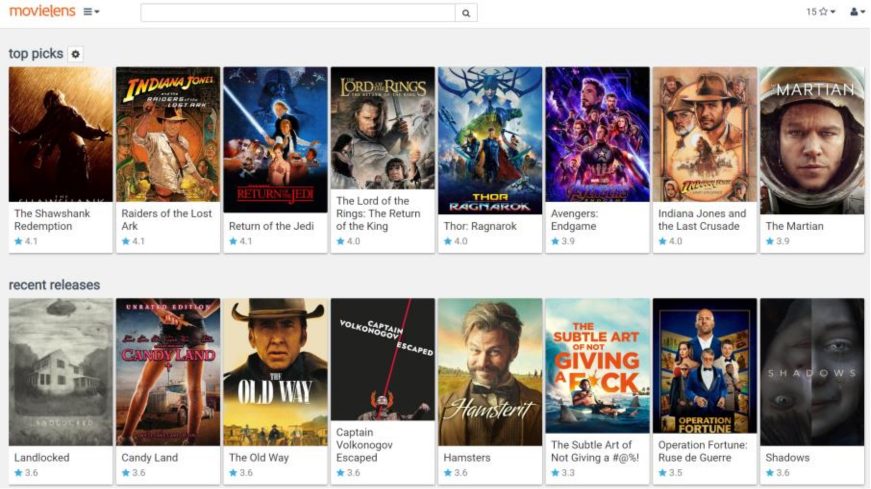
Data

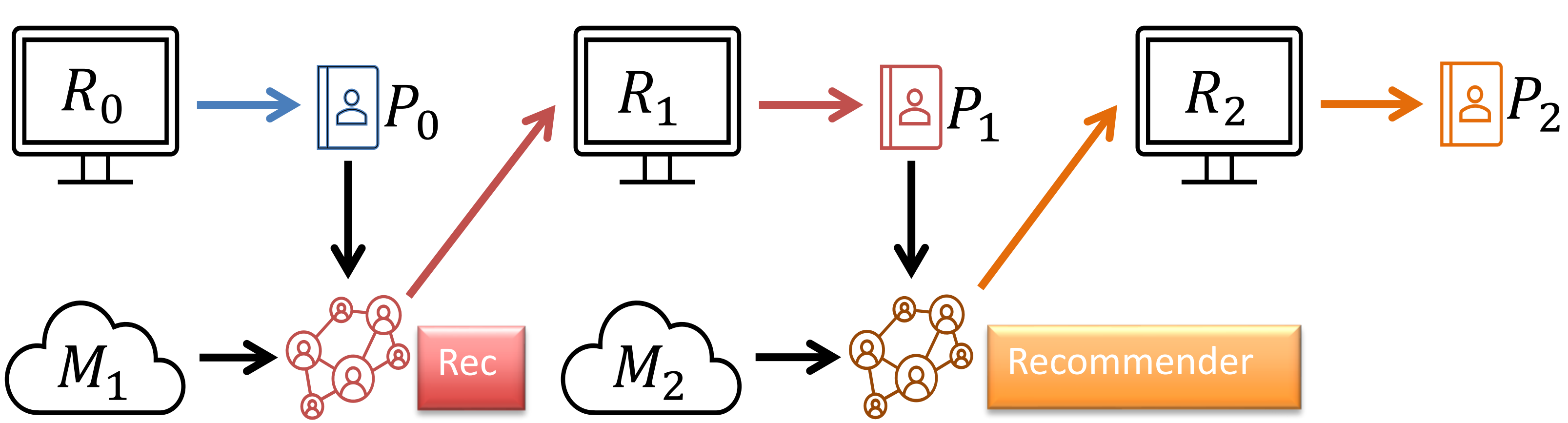
Static dataset

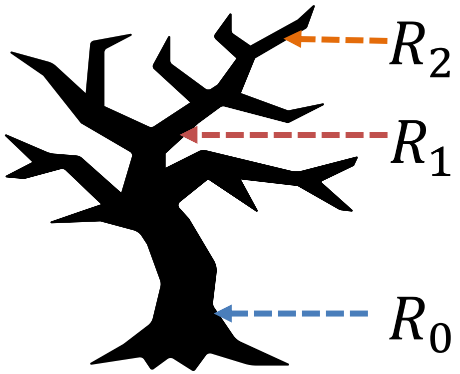
Stream data

The MovieLens dataset

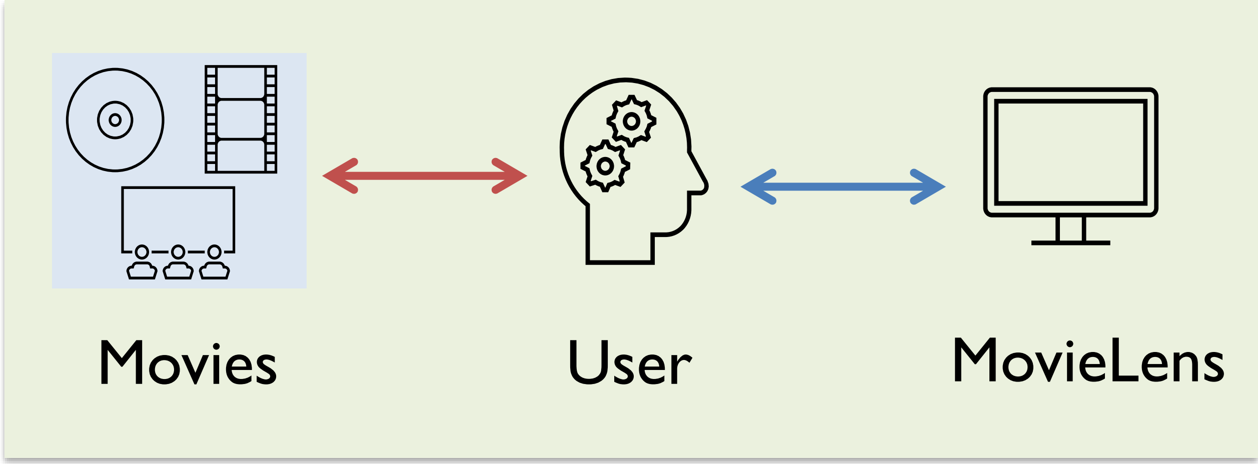






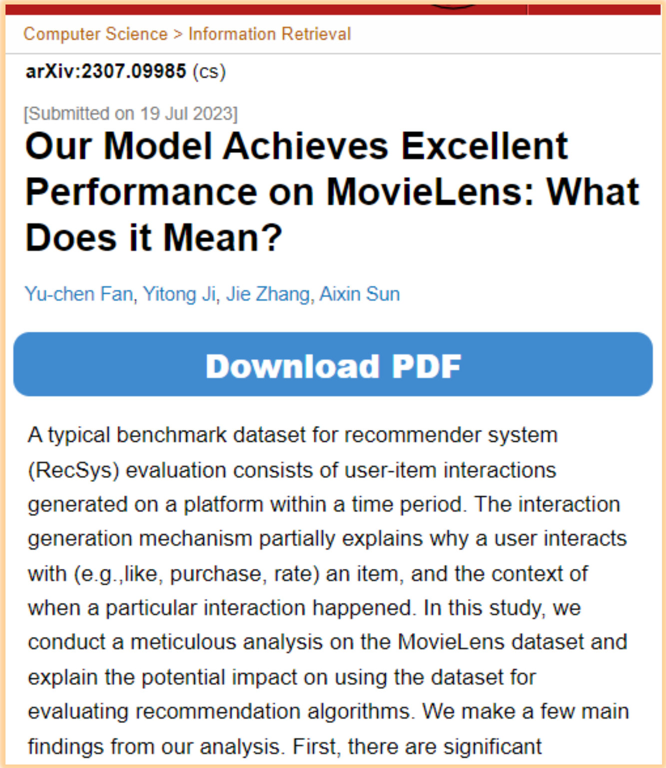


Two Kinds of Interactions



- **User-Movie Interaction**
  - There is a decision process to decide which movie to watch next
- **User-MovieLens Interaction**
  - MovieLens guides users to recall what movies he/she has watched
  - Cold-start dataset for "static preference"

https://arxiv.org/abs/2307.09985


Computer Science > Information Retrieval

arXiv:2307.09985 (cs)

[Submitted on 19 Jul 2023]

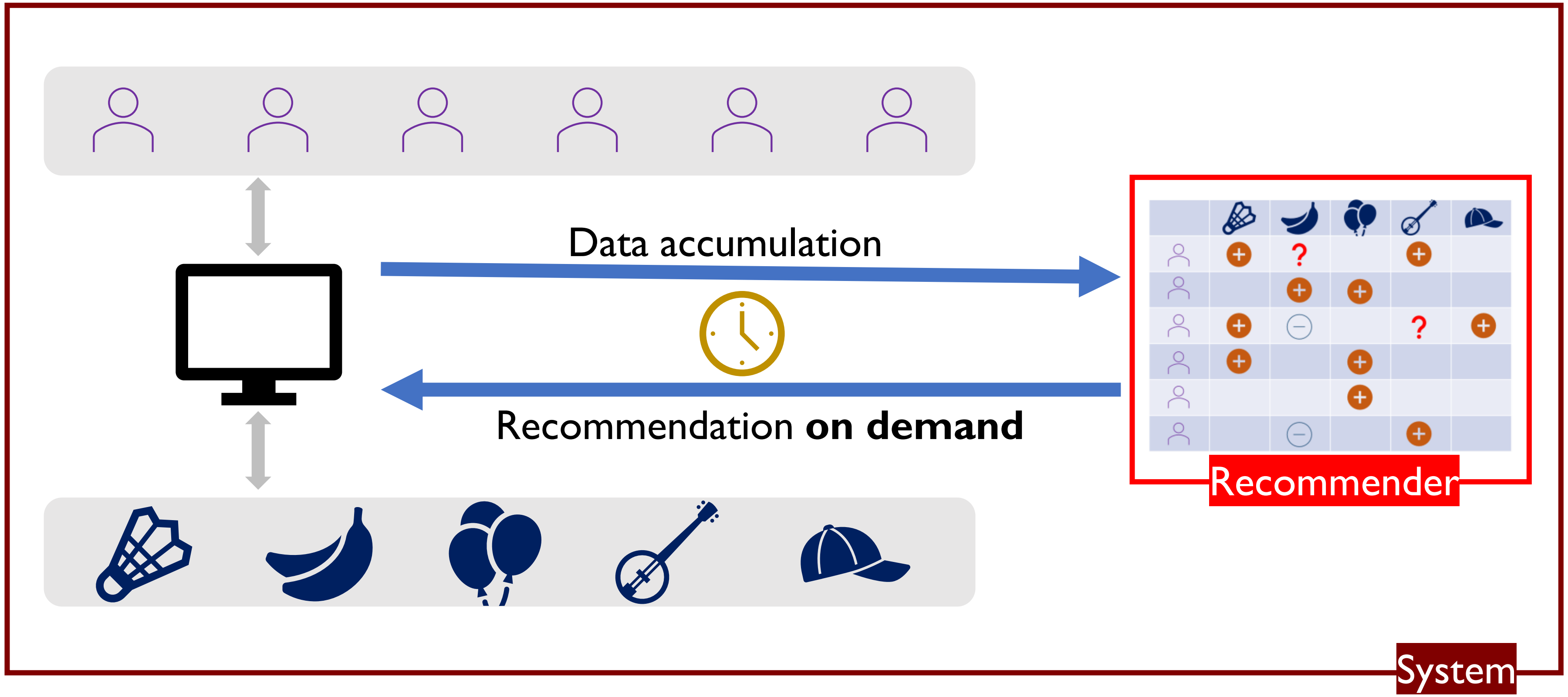
Our Model Achieves Excellent Performance on MovieLens: What Does it Mean?

Yu-chen Fan, Yitong Ji, Jie Zhang, Aixin Sun

Download PDF

A typical benchmark dataset for recommender system (RecSys) evaluation consists of user-item interactions generated on a platform within a time period. The interaction generation mechanism partially explains why a user interacts with (e.g.,like, purchase, rate) an item, and the context of when a particular interaction happened. In this study, we conduct a meticulous analysis on the MovieLens dataset and explain the potential impact on using the dataset for evaluating recommendation algorithms. We make a few main findings from our analysis. First, there are significant

Think about the RecSys problem itself, and its very original research motivation, and not too much on a specific model



Summary

- The original objective of recommender evaluation
  - A **simulation** of the online setting by using an offline dataset
- The importance of observing global timeline
  - A more reliable **simulation** of online setting
  - Minimizing **data leakage**
- The concept of fair evaluation, and user preference modeling
  - Recommenders may choose the **best amount** of data for training
  - User interaction is a **result of decision**
- The selection of dataset
  - A widely used dataset vs some more meaningful datasets

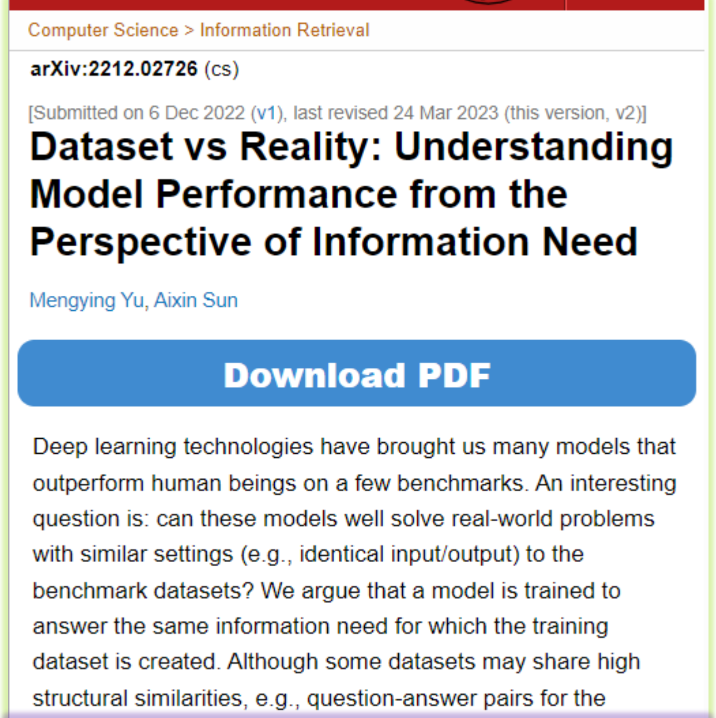
Acknowledgement

Ms. Ji Yitong

Mr. Fan Yu-chen

Dr. Zhang Jie

Dr. Li Chenliang


Computer Science > Information Retrieval

arXiv:2212.02726 (cs)

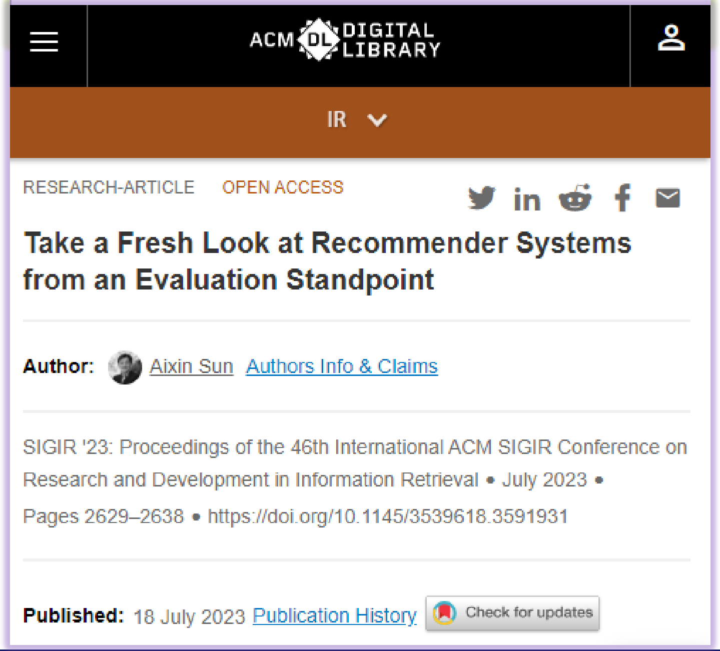
[Submitted on 6 Dec 2022 (v1), last revised 24 Mar 2023 (this version, v2)]

Dataset vs Reality: Understanding Model Performance from the Perspective of Information Need

Mengying Yu, Aixin Sun

Download PDF

Deep learning technologies have brought us many models that outperform human beings on a few benchmarks. An interesting question is: can these models well solve real-world problems with similar settings (e.g., identical input/output) to the benchmark datasets? We argue that a model is trained to answer the same information need for which the training dataset is created. Although some datasets may share high structural similarities, e.g., question-answer pairs for the


ACM DIGITAL LIBRARY

IR

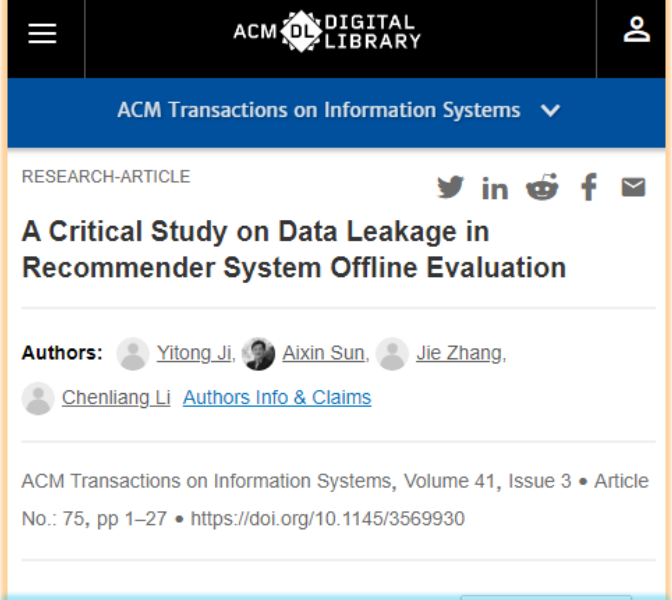
RESEARCH-ARTICLE OPEN ACCESS

Take a Fresh Look at Recommender Systems from an Evaluation Standpoint

Author: Aixin Sun Authors Info & Claims

SIGIR '23: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval • July 2023 • Pages 2629–2638 • https://doi.org/10.1145/3539618.3591931

Published: 18 July 2023 Publication History Check for updates


ACM DIGITAL LIBRARY

ACM Transactions on Information Systems

RESEARCH-ARTICLE

A Critical Study on Data Leakage in Recommender System Offline Evaluation

Authors: Yitong Ji, Aixin Sun, Jie Zhang, Chenliang Li Authors Info & Claims

ACM Transactions on Information Systems, Volume 41, Issue 3 • Article No.: 75, pp 1–27 • https://doi.org/10.1145/3569930


Computer Science > Information Retrieval

arXiv:2307.09985 (cs)

[Submitted on 19 Jul 2023]

Our Model Achieves Excellent Performance on MovieLens: What Does it Mean?

Yu-chen Fan, Yitong Ji, Jie Zhang, Aixin Sun

Download PDF

A typical benchmark dataset for recommender system (RecSys) evaluation consists of user-item interactions generated on a platform within a time period. The interaction generation mechanism partially explains why a user interacts with (e.g.,like, purchase, rate) an item, and the context of when a particular interaction happened. In this study, we conduct a meticulous analysis on the MovieLens dataset and explain the potential impact on using the dataset for evaluating recommendation algorithms. We make a few main findings from our analysis. First, there are significant

https://personal.ntu.edu.sg/axsun/