NLPCC 2019 Tutorials /CCF Advanced Disciplines Lectures (ADL 107), Dunhuang, Gansu, China. Oct. 11, 2019.

Foundations and Trends for Personalized Recommendation

Min Zhang

z-m@tsinghua.edu.cn

Tsinghua University

http://www.thuir.cn/group/~mzhang/publications/NLPCC2019-Tutorial.pdf

Joint contributors: Weizhi Ma, Bin Hao, Hongyu Lu, Shaoyun Shi, Chong Chen, Chenyang Wang

*Images in this slides are mostly from Internet or the original publication.

Outline

Personalized Recommendation:

- Foundations
- Challenges
- Trends

Part I.

Personalized Recommendation: Foundations

Foundations

Personalized Recommendation:

- **Background**
- Tasks
- Information
- Methods
- Evaluation

Background

- Recommender Systems (RS)
  - Nowadays, recommender systems have been deeply integrated into our daily life
  - Videos, news, music, e-commerce, social media, POI, online education, social platforms, …

Background

- Recommender Systems
  - To predict the **user**'s **preference** for an **item** that he/she ~~has never used~~ and when he/she may need it.

Personalized/group recommendation

G. Adomavicius, "Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions", IEEE Transactions on Knowledge and Data Engineering, 2005.

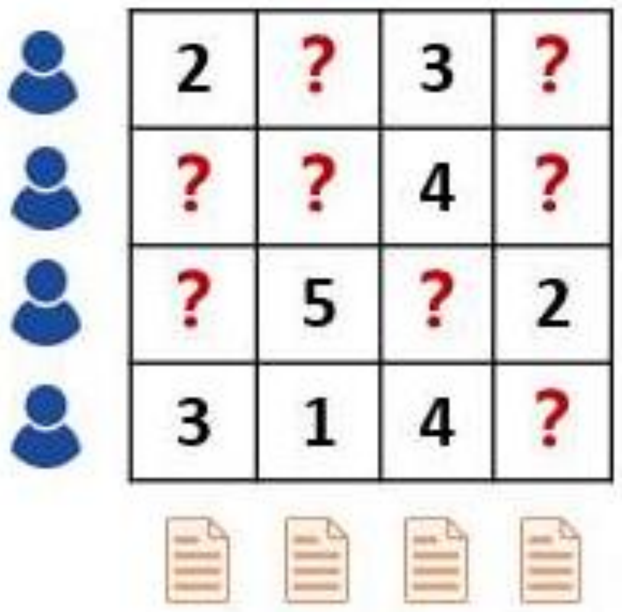
Foundations

Personalized Recommendation:

- Background
- **Tasks**
- Information
- Methods
- Evaluation

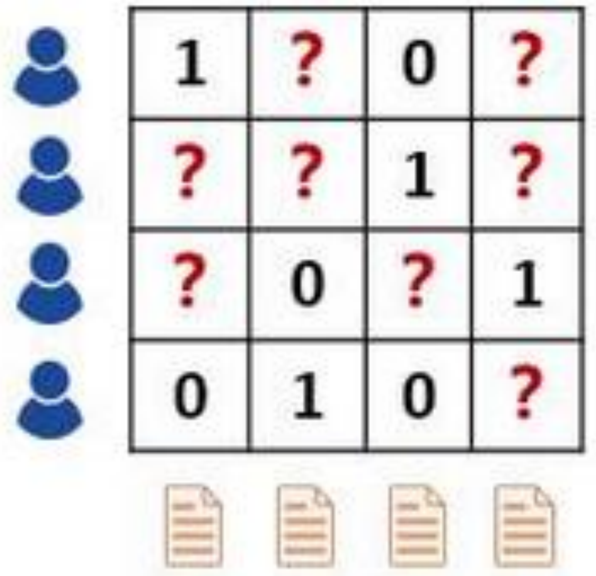
Tasks (1)

- **Rating Prediction**
  - Aiming to **predict** a user's **rating** for those items which were not **rated** yet by him/her.
  - **Predictions** are computed from users' explicit feedback, i.e. their **ratings** provided on some items in the past.



Tasks (2)

- **Click Prediction**
  - Aiming to **predict** the likelihood that something (e.g. an advertisement, a piece of news, a product) will be clicked.
  - **Predictions** are computed from users' implicit feedback, i.e. their **click and purchase history** on some items in the past.



Tasks (3)

- **Top-N Recommendation**
  - Aiming to **identify** a set of *N* items that will be of interest to a certain user .
  - It focuses on the relative preferences of the user for the *N* items.

Foundations

Personalized Recommendation:

- Background
- Tasks
- **Information**
- Methods
- Evaluation

Information in RS

- **User-item Interaction**
  - **Explicit feedback**
    - The user’s explicitly evaluation to items
    - e.g. rating, like/dislike, satisfied, “don’t’ show me again”, etc.
  - **Implicit feedback**
    - Derived from monitoring and analyzing user’s activities
    - E.g. click, view, buy, add to cart, delete,etc
    - Does not require any active user involvement

Information

- **Context**

Multimodal and multi-source information, in general

  - **Textual** e.g. Item categories, item descriptions, user reviews, …
  - **Linking** e.g. Social relations between users, …
  - **Visual** e.g. Item pictures, …
  - **Demographical** e.g. Gender, age, profession, ……
  - **Environment** e.g. OS, platform, desktop/mobile, weather, location, …

Foundations

Personalized Recommendation:

- Background
- Tasks
- Information
- **Methods**
- Evaluation

Methods

- **Collaborative Filtering**
  - Recommended items that people with similar preferences in the past
- **Content-based**
  - Recommended items similar to the ones the user preferred in the past
- **Hybrid**
  - Methods combine collaborative and content-based methods
- **Session-based**
  - To predict the unknown part of a session or the future sessions based on modeling the complex relations in session(s), given partially known session information
- **Neural Models** (for the above categories)

Methods (1): Collaborative Filtering – User-KNN

- K Nearest Neighbors(KNN)
  - The value of the unknown rating $r_{u,i}$ for user *u* and item *i* is usually computed as an aggregate of the ratings of some other (usually the ***N*** most similar) users for the same item *i*.

$$r_{u,i} = \frac{\sum_{u' \in \hat{U}} sim(u, u') \times r_{u',i}}{\sum_{u' \in \hat{U}} |sim(u, u')|}$$

$\hat{U}$ denotes the set of ***N*** users that are the most similar to user ***u*** and who have rated item ***i***

- Similarity between users

$$sim(x, y) = \frac{\sum_{s \in S_{xy}} (r_{x,s} - \bar{r}_x)(r_{y,s} - \bar{r}_y)}{\sqrt{\sum_{s \in S_{xy}} (r_{x,s} - \bar{r}_x)^2 \sum_{s \in S_{xy}} (r_{y,s} - \bar{r}_y)^2}}$$

Pearson Coefficient

$$sim(x, y) = \cos(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{\|\vec{x}\|_2 \times \|\vec{y}\|_2} = \frac{\sum_{s \in S_{xy}} r_{x,s} r_{y,s}}{\sqrt{\sum_{s \in S_{xy}} r_{x,s}^2} \sqrt{\sum_{s \in S_{xy}} r_{y,s}^2}}$$

Cosine Similarity

[G. Adomavicius, and A. Tuzhilin, ... Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. TKDE’2005]

Methods (1): Collaborative Filtering – Item-KNN

- The method computes the prediction of an item *i* for a user *u* by computing the sum of the ratings given by the user on the items similar to *i* (usually the ***N*** most similar).

$$P_{u,i} = \frac{\sum_{\text{all similar items, N}} (s_{i,N} * R_{u,N})}{\sum_{\text{all similar items, N}} (|s_{i,N}|)}$$

Cosine Similarity

$$sim(i,j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{||\vec{i}||_2 * ||\vec{j}||_2}$$

Correlation-based Similarity

$$sim(i,j) = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_i)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_j)^2}}.$$

Adjusted Cosine Similarity

$$sim(i,j) = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_u)(R_{u,j} - \bar{R}_u)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_u)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_u)^2}}$$

Sarwar B, Karypis G, Konstan J, et al. Item-based Collaborative Filtering Recommendation Algorithms[C], International Conference on World Wide Web. 2001.

Methods (1): Collaborative Filtering – SVD

Information Retrieval @ Tsinghua University

- Matrix Factorization
  - Singular Value Decomposition (SVD)

Basic Model: $\hat{r}_{ui} = q_i^T p_u$

Adding Biases: $\hat{r}_{ui} = \mu + b_i + b_u + q_i^T p_u$

Adding Temporal: $\hat{r}_{ui}(t) = \mu + b_i(t) + b_u(t) + q_i^T p_u(t)$

2	?	3	?
?	?	4	?
?	5	?	2
3	1	4	?

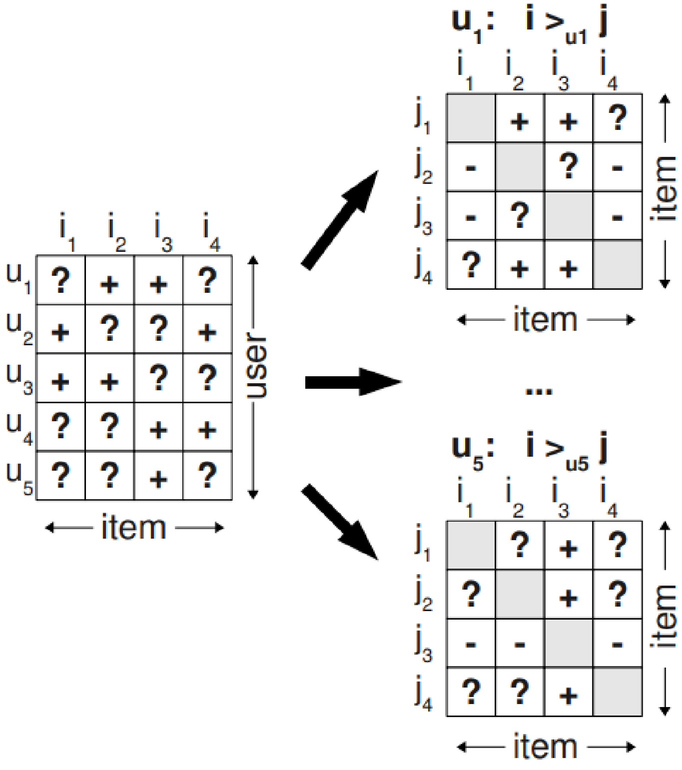
[Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. Computer, 2009]

Methods (1): Collaborative Filtering – BPR

- Bayesian Personalized Ranking (BPR)

$$\hat{r}_{ui} = {q_i}^T p_u = \sum_{f=1}^{k} p_{uf} \cdot q_{if}$$

$$\hat{r}_{uij} = \hat{r}_{ui} - \hat{r}_{uj}$$



BPRMF tries to classify the difference of the two predictions

$$\frac{\partial}{\partial\theta}\hat{r}_{uij} = \begin{cases} (q_{if} - q_{jf}) & if \quad \theta = p_{uf} \\ p_{uf} & if \quad \theta = q_{if} \\ -p_{uf} & if \quad \theta = q_{jf} \\ 0 & else \end{cases}$$

[Rendle S, et al. BPR: Bayesian personalized ranking from implicit feedback. UAI 2009.]

Methods

- **Collaborative Filtering**
  - Recommended items that people with similar preferences in the past
- **Content-based**
  - Recommended items similar to the ones the user preferred in the past
- **Hybrid**
  - Methods combine collaborative and content-based methods
- **Session-based**
  - To predict the unknown part of a session or the future sessions based on modeling the complex relations in session(s), given partially known session information
- **Neural Models** (for the above categories)

Content-based (1) : Content Similarity

- Score Function

$$s(u,i) = score(Profile(u), Attributes(i))$$

  - E.G. (When recommending web pages, URLs, documents)

$$s(u,i) = cos(\vec{w_u}, \vec{w_i}) = \frac{\vec{w_u} \cdot \vec{w_i}}{\|\vec{w_u}\|_2 \times \|\vec{w_i}\|_2} = \frac{\sum_{k=1}^{d} w_{u,k} w_{i,k}}{\sqrt{\sum_{k=1}^{d} w_{u,k}^2} \sqrt{\sum_{k=1}^{d} w_{i,k}^2}}$$

$\vec{w_u}$ and $\vec{w_i}$ are TF-IDF vectors

$$TF_{i,j} = \frac{f_{i,j}}{\max_z f_{z,j}} \qquad IDF_i = \log \frac{N}{n_i} \qquad w_{i,j} = TF_{i,j} \times IDF_i$$

Content based (2): Factorization Machine

Motivation:

First-order: Linear Regression   Second-order: pair-wise interactions between non-zero features

$$\tilde{y}(x) = w_0 + \overbrace{\sum_{i=1}^{n} w_i x_i}^{\text{First-order}} + \overbrace{\sum_{i=1}^{n} \sum_{j=i+1}^{n} \boxed{w_{ij}} x_i x_j}^{\text{Second-order}}$$

$W = \{w_{ij}\}_{n\times n}$ brings too much parameters: $O(n^2)$.
Some pairs rarely occur: w_{ij} hard to train.

Input x:

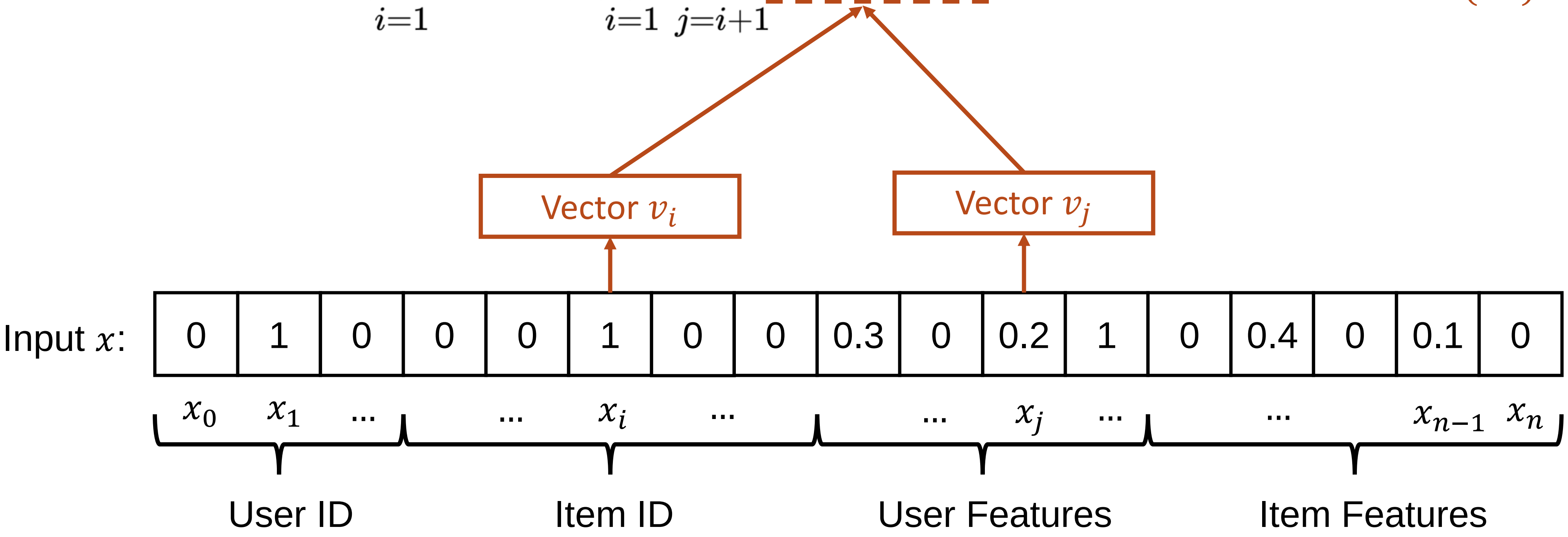
0	1	0	0	0	1	0	0	0.3	0	0.2	1	0	0.4	0	0.1	0
x_0	x_1	…		…	x_i	…			…	x_j	…		…		x_{n-1}	x_n

$\underbrace{x_0 \; x_1 \; \dots}_{\text{User ID}} \; \underbrace{\dots \; x_i \; \dots}_{\text{Item ID}} \; \underbrace{\dots \; x_j \; \dots}_{\text{User Features}} \; \underbrace{\dots \; x_{n-1} \; x_n}_{\text{Item Features}}$

Content based (2): Factorization Machine

$$\tilde{y}(x) = w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \boxed{< v_i, v_j >} x_i x_j$$

Replace w_{ij} by dot product of two vectors v_i and v_j with dimension d. $O(dn)$

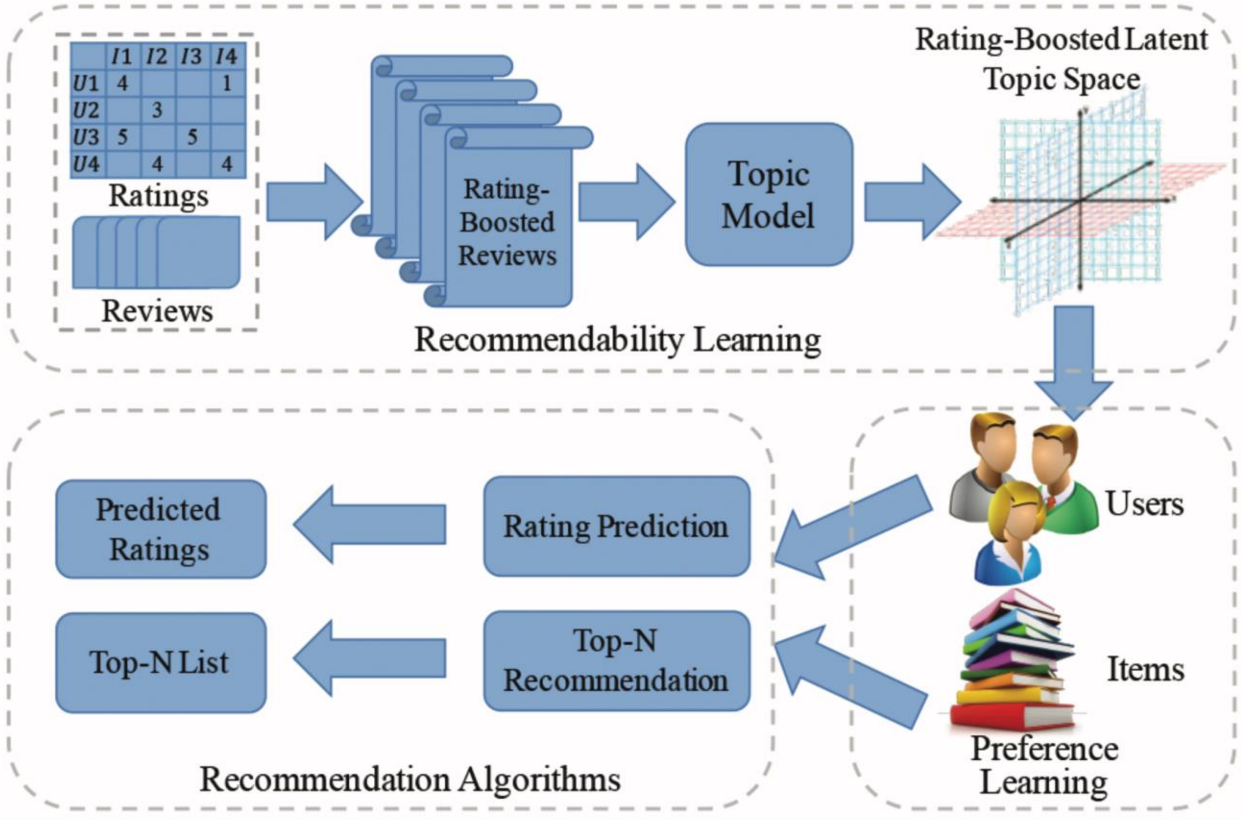


Methods

- **Collaborative Filtering**
  - Recommended items that people with similar preferences in the past
- **Content-based**
  - Recommended items similar to the ones the user preferred in the past
- **Hybrid**
  - Methods combine collaborative and content-based methods
- **Session-based**
  - To predict the unknown part of a session or the future sessions based on modeling the complex relations in session(s), given partially known session information
- **Neural Models** (for the above categories)

Hybrid (1): Incorporating User Reviews

- Main idea: introducing user reviews into recommendation



$$\hat{r}_{u,i} = \mu + b_u + b_i + (q_u + v_u)(p_i + f_i)^T$$

v_u is preference distribution of user u and f_i is the recommendability distribution of item i. They are both extracted from reviews.

[Tan Y, Zhang M, Liu Y, et al. Rating-boosted latent topics: Understanding users and items with ratings and reviews. IJCAI'2016]

Experimental Results

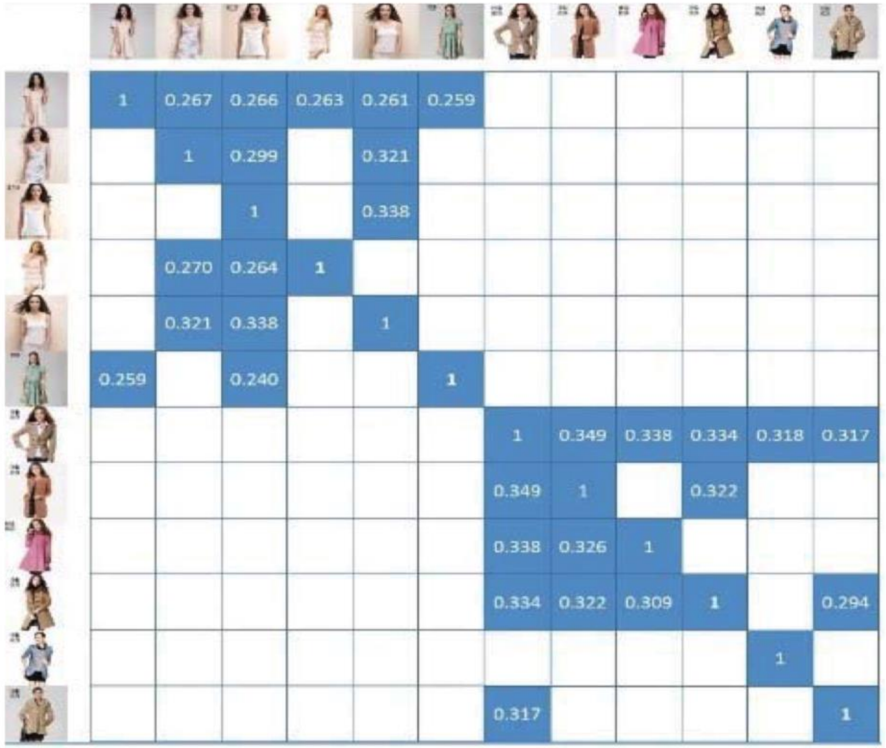
Dataset	LFM (a)	HFT (b)	CTR (c)	RMR (d)	HFT+RB (e)	no-RBLT (f)	RBLT (g)
Arts	1.363	1.388	1.471	1.371	1.362	1.375	**1.352**
Automotive	1.431	1.428	1.492	**1.403**	1.419	1.430	1.406
Baby	1.596	1.631	N/A	N/A	1.619	1.618	**1.583**
Beauty	1.375	1.347	1.361	1.334	1.369	1.361	**1.308**
Cell Phones & Accessories	2.124	2.129	2.177	**2.085**	2.135	2.125	2.101
Clothing & Accessories	0.398	**0.327**	0.355	0.336	0.358	0.378	0.328
Electronics	1.670	1.724	1.764	1.722	1.695	1.665	**1.665**
Gourmet Foods	1.439	1.431	1.482	1.465	1.446	1.441	**1.428**
Health	1.503	1.528	1.552	1.512	1.494	1.513	**1.479**
Home & Kitchen	1.521	1.527	1.577	**1.501**	1.544	1.548	1.533
Industrial & Scientific	0.387	0.357	0.382	0.362	0.377	0.370	**0.353**
Jewelry	1.209	1.178	1.206	1.160	1.171	1.198	**1.146**
Kindle_Store	**1.390**	1.421	1.457	1.412	1.419	1.393	1.394
Movies	0.456	1.119	1.114	1.120	0.710	0.382	**0.359**
Music	0.707	0.969	0.959	0.959	0.856	0.729	**0.625**
Musical Instruments	1.430	1.396	1.422	**1.374**	1.411	1.438	1.444
Office Products	1.613	1.680	1.733	1.638	1.603	1.620	**1.579**
Patio	1.686	1.708	1.720	**1.669**	1.707	1.684	1.680
Pet Supplies	**1.544**	1.582	1.613	1.562	1.569	1.548	1.556
Shoes	0.293	0.226	0.271	0.251	0.224	0.262	**0.209**
Software	2.218	2.197	2.254	2.173	2.197	2.248	**2.151**
Sports & Outdoors	1.153	1.136	1.150	**1.129**	1.169	1.142	1.132
Tools & Home Impro.	1.489	1.499	1.513	1.491	1.495	1.470	**1.465**
Toys & Games	1.372	1.366	1.389	1.372	**1.362**	1.369	1.365
Video Games	1.487	1.511	1.572	1.510	1.482	1.504	**1.462**
Watches	1.497	1.486	1.491	**1.458**	1.485	1.493	1.487
Average	1.321	1.357	1.379	1.335	1.334	1.319	1.292

sound	device	battery	surface	camera
headset	music	battery	case	camera
ear	bluetooth	charge	clip	phone
headphone	device	price	belt	screen
sound	treo	new	screen	nokia
volume	ipod	cable	cover	picture
comf.	keyboard	cell	nice	feature
easy	software	days	leather	cell
noise	player	original	color	reception
motorola	cable	item	plastic	motorola
unit	palm	service	price	easy

Table 3: Top ten words of each topic discovered by RBLT from *Cell Phones & Accessories*. Each column is corresponding to a topic attached with an "interpretation" label.

Hybrid (2): Incorporating Item Pictures

- Main idea: items' visual similarity can be helpful for the recommendation



- Items with a higher visual similarity should hold a shorter distance of implicit features, so:

$$\min_{V}\left\{\lambda_S \sum_{\substack{i,j\in Z\\ i,j\in[1,N]}} S_{ij}\left\|v_i - v_j\right\|_F^2\right\}$$

- Using matrix supplement, reconstruct user matrix C and item matrix V as:

$$P = [C\ H_1],\quad Q = [V\ H_2]$$

- Objective function:

$$\min_{P,Q}\left\{\left\|PQ^T - D\right\|_F^2 + \lambda_S \sum_{\substack{i,j\in Z\\ i,j\in[1,N]}} S_{ij}\left\|v_i - v_j\right\|_F^2 + \lambda_C\|C\|_F^2 + \lambda_V\|V\|_F^2 + \lambda_H\left(\|H_1\|_F^2 + \|H_2\|_F^2\right)\right\}$$

[C Guo, et al. A Picture is Worth A Thousand Words: Introducing Visual Similarity into Recommendation. ICICIP'2016]

Experimental Results

TABLE V. PROPERTY OF THE MODELS

	Constrained by Visual similarity	Using implicit factors	With matrix supplement
Base	No	No	No
Item-based	No	No	No
CF	No	No	No
CF-CV	Yes	No	No
CF-CV-I	Yes	Yes	No
CF-MS	No	No	Yes
CF-MS-CV	Yes	No	Yes
CF-MS-CV-I	**Yes**	**Yes**	**Yes**

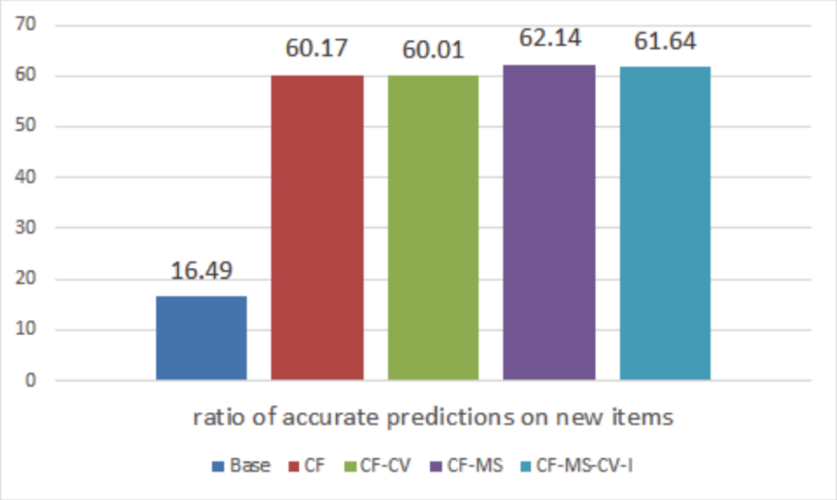


Fig. 5. Ratio of accurate predictions on new items

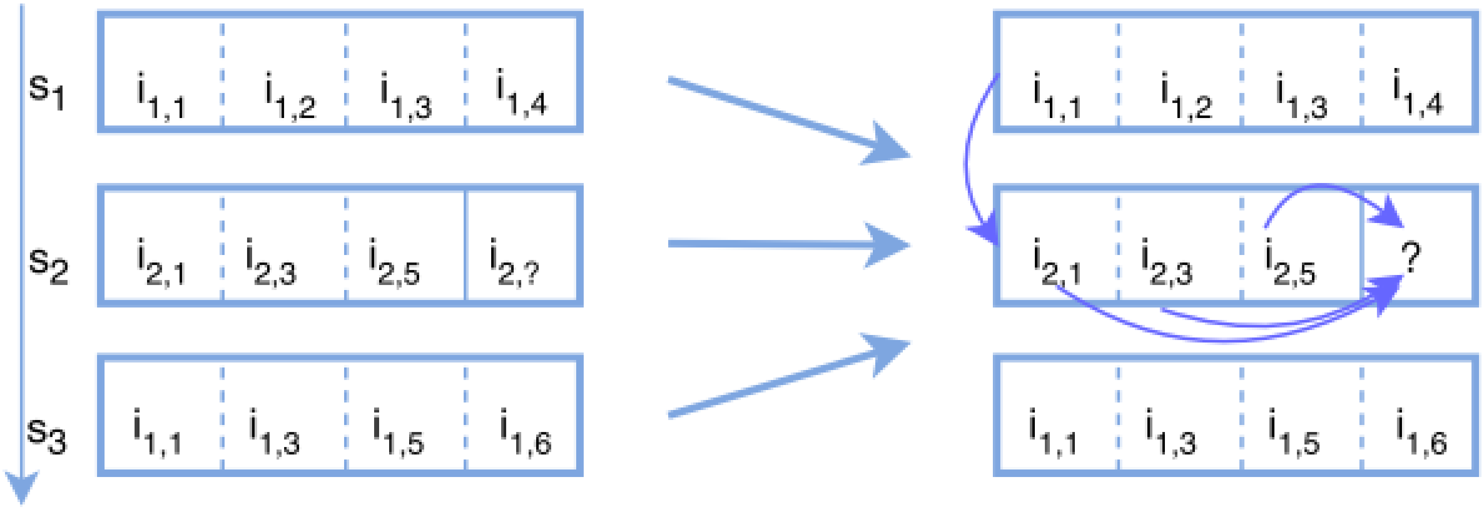
TABLE VI. RMSE WITH DIFFERENT MODELS

	Base	Item-based	CF	CF-CV	CF-CV-I	CF-MS	CF-MS-CV	CF-MS-CV-I
Mixed	0.6346	0.6010	0.5887	0.5878	0.5879	0.5775	0.5776	**0.5688**
Cold start	0.6698	**0.6542**	0.7158	0.7620	0.7311	**0.6549**	0.6573	0.6555

Methods

- **Collaborative Filtering**
  - Recommended items that people with similar preferences in the past
- **Content-based**
  - Recommended items similar to the ones the user preferred in the past
- **Hybrid**
  - Methods combine collaborative and content-based methods
- **Session-based**
  - To predict the unknown part of a session or the future sessions based on modeling the complex relations in session(s), given partially known session information
- **Neural Models** (for the above categories)

Session based Recommendation(1)

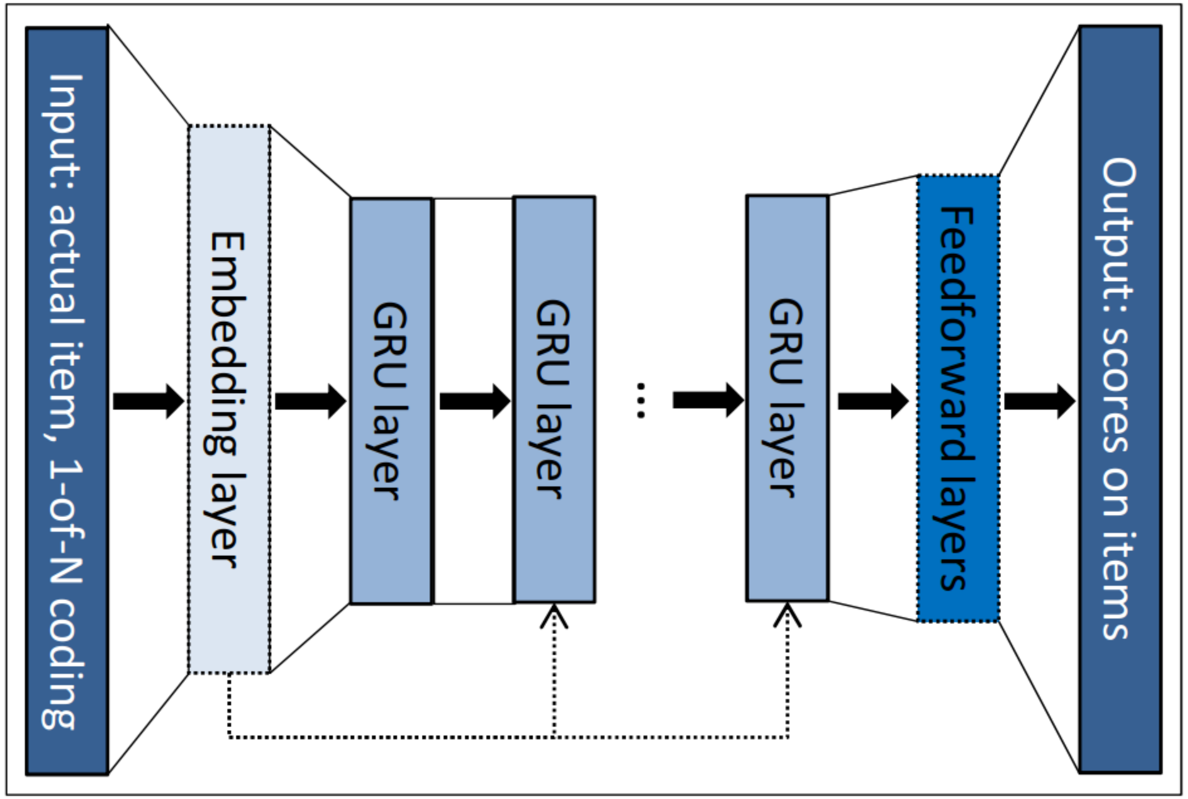


Aim to predict the missing items by modeling the intra- and inter-session dependency

[Shoujin Wang, Longbing Cao, Yan Wang. A Survey on Session-based Recommender Systems. arXiv'2019.]

Session based Recommendation(2)

- GRU4Rec



[Hidasi B, Karatzoglou A, Baltrunas L, et al. Session-based recommendations with recurrent neural networks. ICLR 2016.]

Experimental Results

Baseline	RSC15		VIDEO	
	Recall@20	MRR@20	Recall@20	MRR@20
POP	0.0050	0.0012	0.0499	0.0117
S-POP	0.2672	0.1775	0.1301	0.0863
Item-KNN	0.5065	0.2048	0.5508	0.3381
BPR-MF	0.2574	0.0618	0.0692	0.0374

Table 3: Recall@20 and MRR@20 for different types of a single layer of GRU, compared to the best baseline (item-KNN). Best results per dataset are highlighted.

Loss / #Units	RSC15		VIDEO	
	Recall@20	MRR@20	Recall@20	MRR@20
TOP1 100	0.5853 (+15.55%)	0.2305 (+12.58%)	0.6141 (+11.50%)	0.3511 (+3.84%)
BPR 100	0.6069 (+19.82%)	0.2407 (+17.54%)	0.5999 (+8.92%)	0.3260 (-3.56%)
Cross-entropy 100	0.6074 (+19.91%)	0.2430 (+18.65%)	0.6372 (+15.69%)	0.3720 (+10.04%)
TOP1 1000	0.6206 (+22.53%)	**0.2693 (+31.49%)**	**0.6624 (+20.27%)**	**0.3891 (+15.08%)**
BPR 1000	**0.6322 (+24.82%)**	0.2467 (+20.47%)	0.6311 (+14.58%)	0.3136 (-7.23%)
Cross-entropy 1000	0.5777 (+14.06%)	0.2153 (+5.16%)	–	–

Methods

- **Collaborative Filtering**
  - Recommended items that people with similar preferences in the past
- **Content-based**
  - Recommended items similar to the ones the user preferred in the past
- **Hybrid**
  - Methods combine collaborative and content-based methods
- **Session-based**
  - To predict the unknown part of a session or the future sessions based on modeling the complex relations in session(s), given partially known session information
- **Neural Models** (for the above categories)

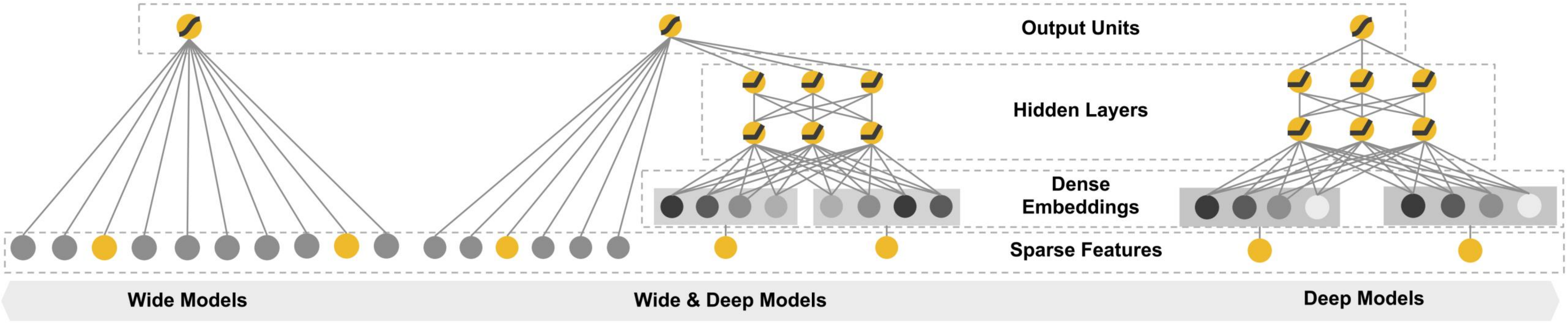
Why Neural Models?

- **Nonlinear Transformation**
  - Capture complex and intricate user/item interaction patterns.
- **Representation Learning**
  - Process heterogeneous information.
- **Sequence Modeling**
  - Next-item/basket prediction and session-based recommendation.
- **Flexibility**
  - Easily build hybrid and composite models.

[Zhang S, Yao L, Sun A, et al. Deep learning based recommender system: A survey and new perspectives. CSUR 2019]

Neural models (1): Wide & Deep

- Memorization (Wide) and Generalization (Deep)



[Cheng H T, Koc L, Harmsen J, et al. Wide & deep learning for recommender systems. The 1st workshop on deep learning for recommender systems. 2016]

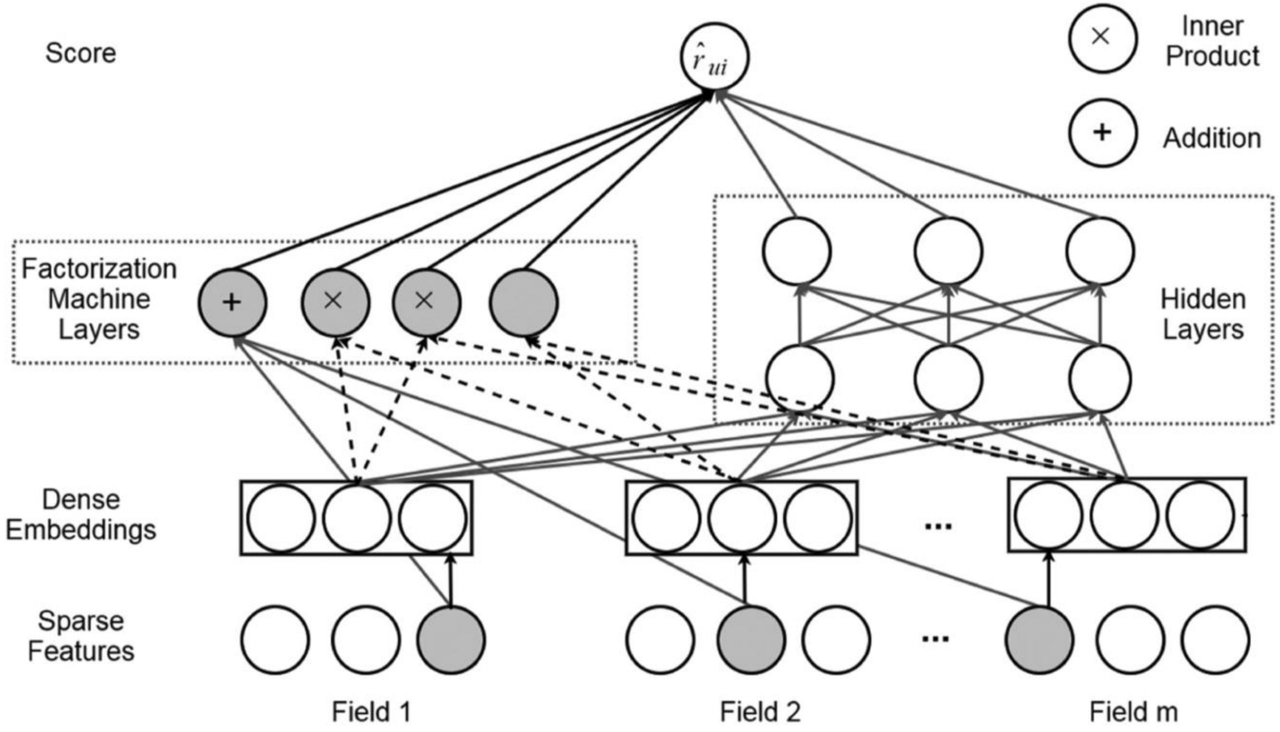
Experimental Results

- Combination achieves statistically significant improvements.

Model	Offline AUC	Online Acquisition Gain
Wide (control)	0.726	0%
Deep	0.722	+2.9%
Wide & Deep	0.728	+3.9%

Neural models (2): Deep Factorization Machine

- Automatically capture high-order feature interactions.



[Guo H, Tang R, Ye Y, et al. DeepFM: a factorization-machine based neural network for CTR prediction. IJCAI 2017]

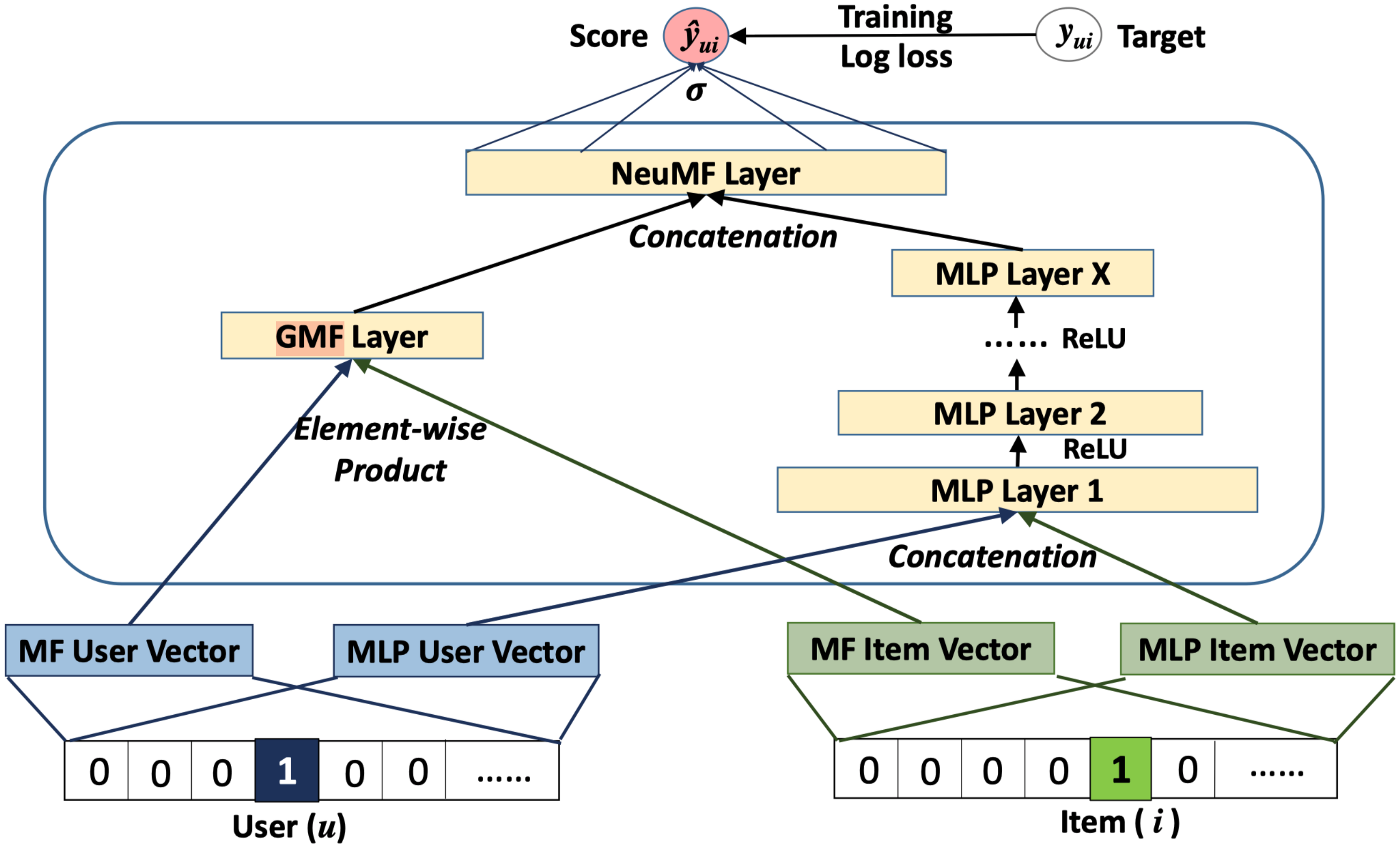
Experimental Results

- DeepFM outperforms Wide&Deep

	Company∗		Criteo	
	AUC	LogLoss	AUC	LogLoss
LR	0.8641	0.02648	0.7804	0.46782
FM	0.8679	0.02632	0.7894	0.46059
FNN	0.8684	0.02628	0.7959	0.46350
IPNN	0.8662	0.02639	0.7971	0.45347
OPNN	0.8657	0.02640	0.7981	0.45293
PNN∗	0.8663	0.02638	0.7983	0.45330
Wide&Deep: LR & DNN	0.8671	0.02635	0.7858	0.46596
FM & DNN	0.8658	0.02639	0.7980	0.45343
DeepFM	**0.8715**	**0.02619**	**0.8016**	**0.44985**

Neural models (3): Neural Collaborative Filtering

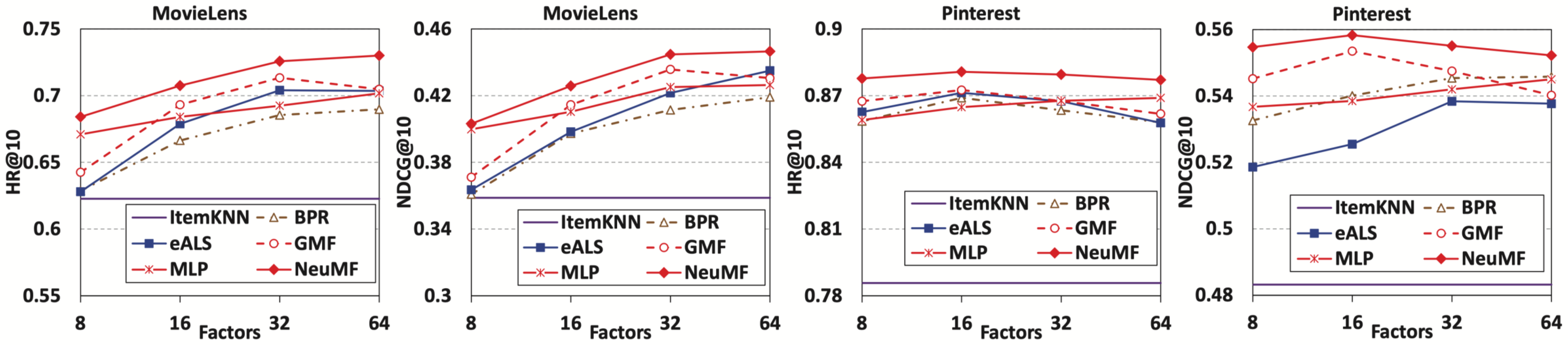
- NeuMF



[He X, Liao L, Zhang H, et al. Neural collaborative filtering. WWW 2017]

Experimental Results

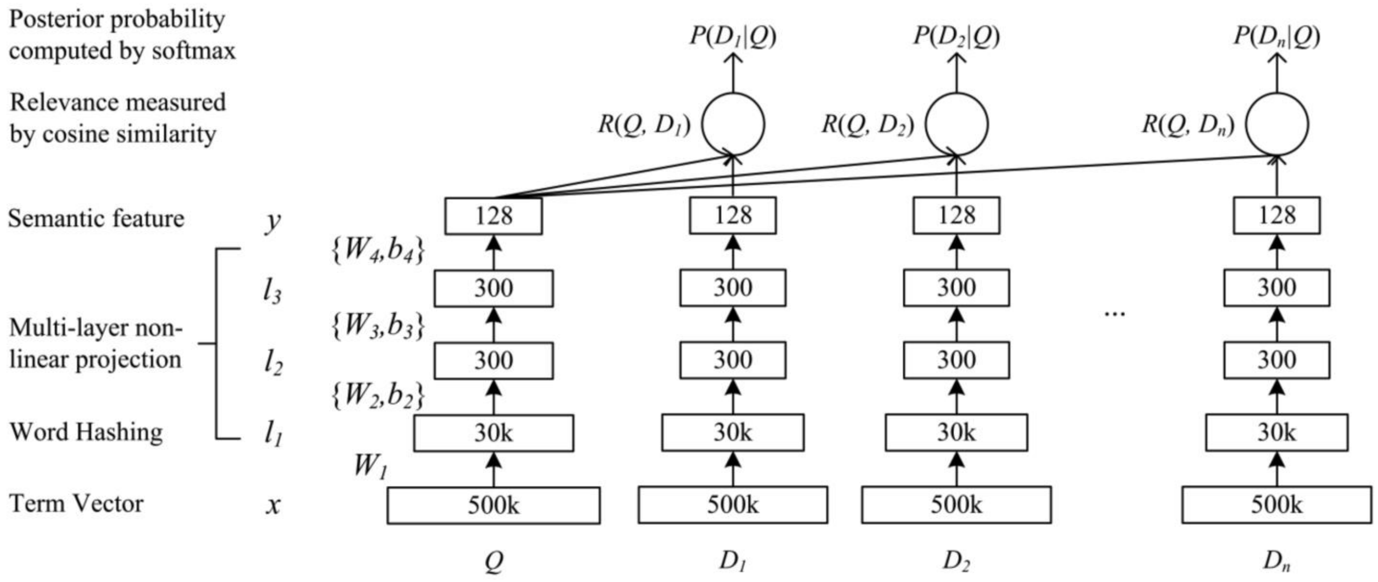
- NeuMF outperforms GMF, ItemKNN, BPR, etc

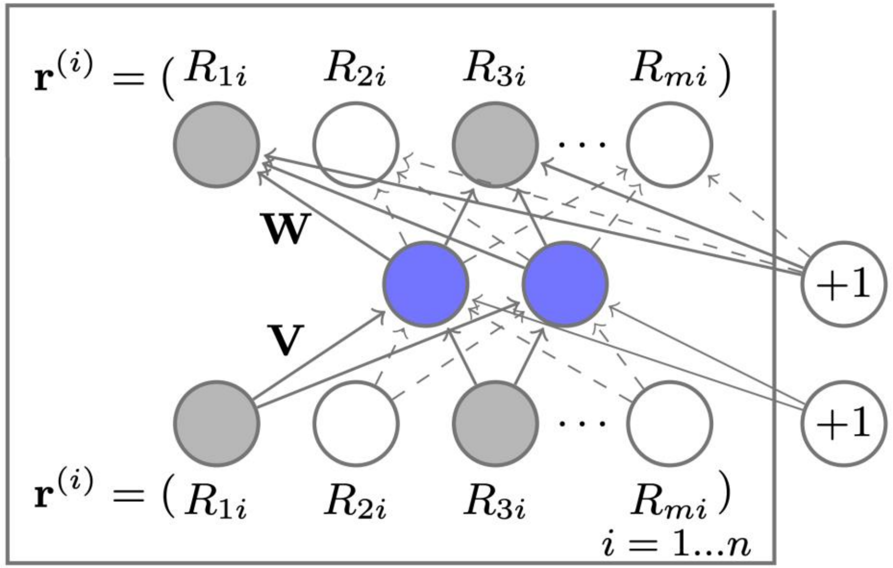


(a) MovieLens — HR@10 (b) MovieLens — NDCG@10 (c) Pinterest — HR@10 (d) Pinterest — NDCG@10

Neural models (4): Other Deep Models

- DSSM (Deep Structured Semantic Model)
- Autorec (Autoencoders & CF)





Huang P S, He X, Gao J, et al. Learning deep structured semantic models for web search using clickthrough data. CIKM 2013

Sedhain S, Menon A K, Sanner S, et al. AutoRec: Autoencoders Meet Collaborative Filtering. WWW 2015

Foundations

Personalized Recommendation:

- Background
- Tasks
- Information
- Methods
- **Evaluation**

Evaluation – Methodology(1)

- **Offline Evaluation**
  - Definition
    - Measured based on static historical data (log, e-commerce review)
  - Different Dataset Splitting methodology (training, validation, test)
    - Proportional splitting (e.g. 8/1/1)
    - Cross validation (e.g. 5 fold, 10 fold)
    - Sampling: 1: x (e.g. 1:99)
    - Full: 1:All
    - Leave-one-out

Evaluation – Methodology(2)

- **Online Evaluation**
  - Definition

    Design an online experiment and measure the performance of the recommender system based on users' online feedback results
  - A/B Test
    - A/B Test is to develop two schemes to achieve the same goal. Some users use Plan A and others use Plan B to record the feedback of the two group of users, and then confirm which scheme is better according to the corresponding evaluation metrics

Evaluation - Metrics

- Metrics
  - Rating Prediction

$$RMSE = \sqrt{\frac{\sum_{(u,i)\epsilon E^U}(y_{ui} - \hat{y}_{ui})^2}{|E^U|}} \qquad MAE = \frac{\sum_{(u,i)\epsilon E^U}|y_{ui} - \hat{y}_{ui}|}{|E^U|}$$

  - Click/Purchase Prediction
    - AUC (Area under curve)
  - Top-N Recommendation

$$NDCG@ = Z_k \sum_{i=1}^{K} \frac{2^{r_i}-1}{log_2(i+1)} \qquad HR@K = \frac{NumberofHits@K}{|GT|}$$

Part II.

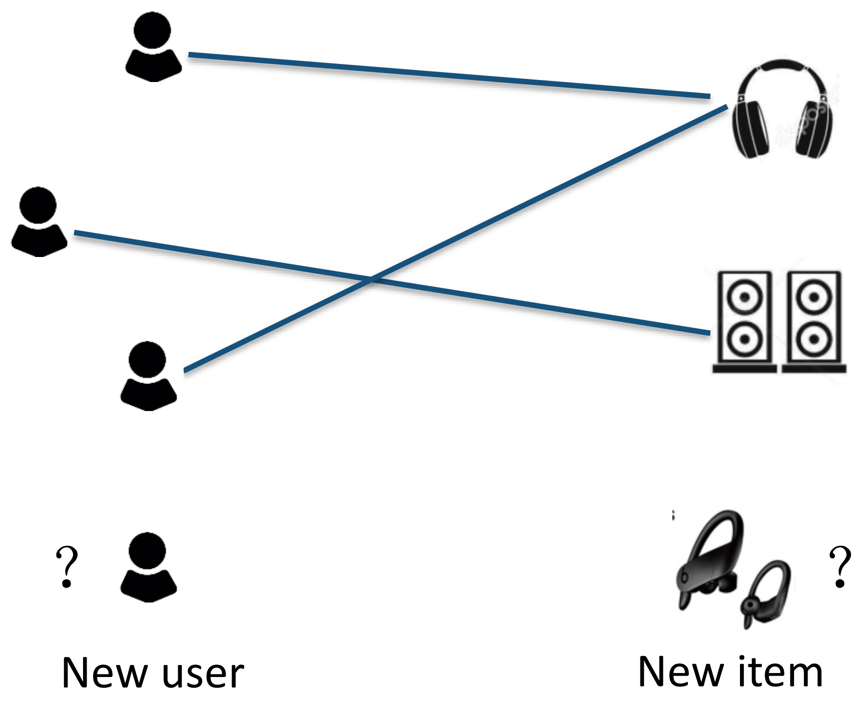
Personalized Recommendation: Challenges

Challenges

- Cold-start
- Efficiency
- Explainability
- User satisfaction & behavior
- Exploration vs. exploitation
- Fairness

Cold-start

- How to deal with users or items with less (or no) interaction history?



Efficiency

- Millions (even billions) of users and items in real systems, and real-time recommendation is necessary in many scenarios
- Efficiency recommender system is needed!
  - Less parameters
  - Shorter training time
  - Lower computational cost
  - Easy to update

Explainability

For users

- Why did you show this result to me?
  - Recommend this product
  - Show this piece of news

For systems/engineers

- Why does my system give this output?
- Where do they (errors, bonus) come from?
- What factor(s) are the most important one(s)?

Can we find some solutions that are both highly *accurate* and easily *explainable*?

User Satisfaction & Behavior

- Why user satisfaction is an important metric?
  - User satisfied → User will come back again
  - User satisfaction → Real preference
- Evaluation metrics so far:
  - Click Through Rate (CTR)
  - View time
  - …

Do these metrics can reflect users' preference

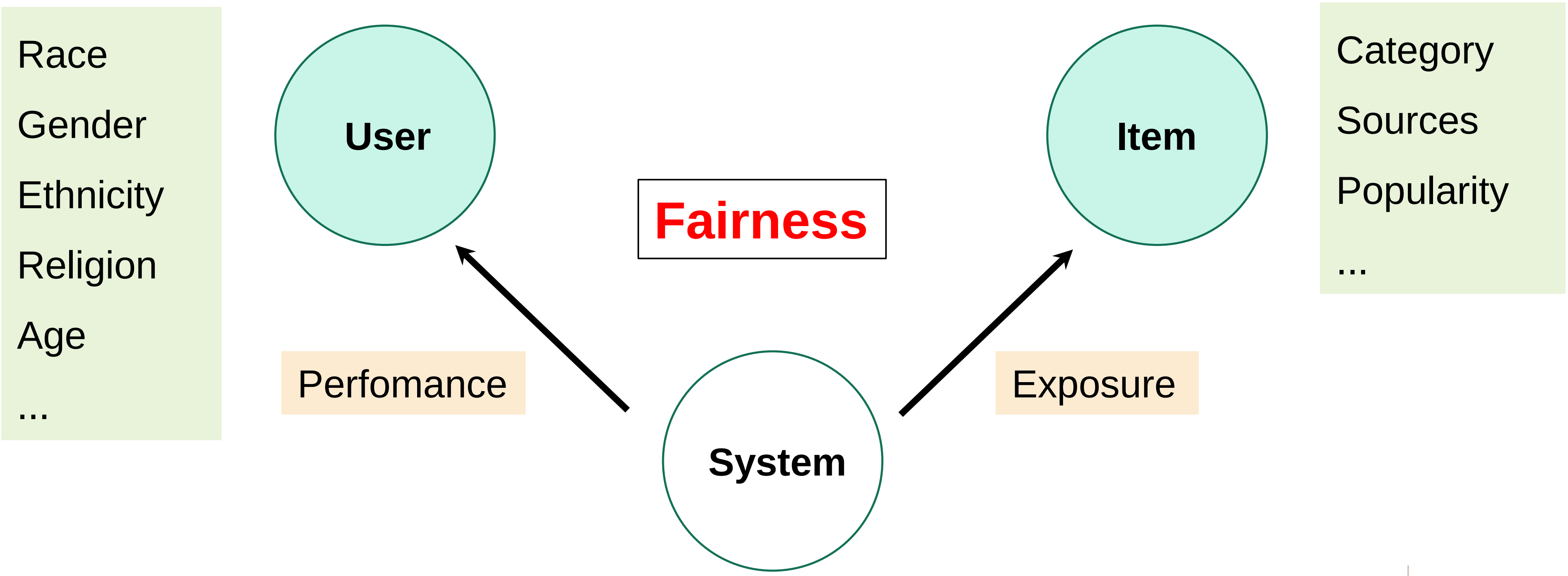
Exploration vs. Exploitation

- Diversity: the diversity of items in the recommendation list, which has the contrary effect of similarity.
  - Balancing recommendation lists to cover the user's whole set of interests.
  - Represents variety of items in recommendation lists

What if the status is:
"After purchased this item?

Fairness

“A person‘s experience with an information system should irrelevantly depend on their personal characteristics (Michael et al. 2018)"



Challenges: overview

- Cold-start
- Efficiency
- Explainability
- User satisfaction & behavior
- Exploration vs. exploitation
- Fairness

Part III.

Personalized Recommendation:

Trends

Trends

- Towards challenges
  - **Cold-start**
  - Efficiency
  - Explainability
  - User satisfaction & behavior
  - Exploration vs. exploitation (Diversity)
  - Fairness

- Trending techniques
  - Knowledge-aware
  - Sequential Recommendation
  - Reinforcement Learning
- New scenarios

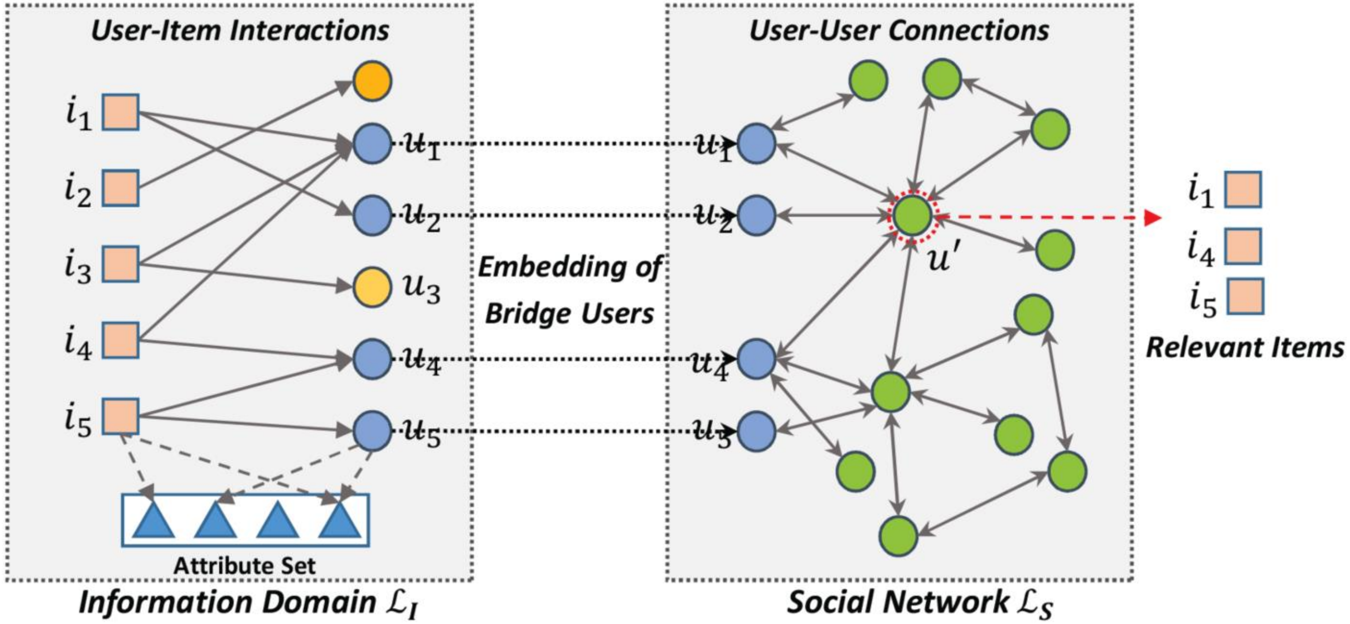
Towards Challenges – Cold-Start

Three strategies:

- **Investigating more content Information**
  - Item Content
  - User Profile
- **Leveraging cross Platform/Domain**
  - Social Information
  - Multimedia
- **Building hybrid Models**
  - Adjust strategies in different scenarios

1. Using Cross Domain Information (1)

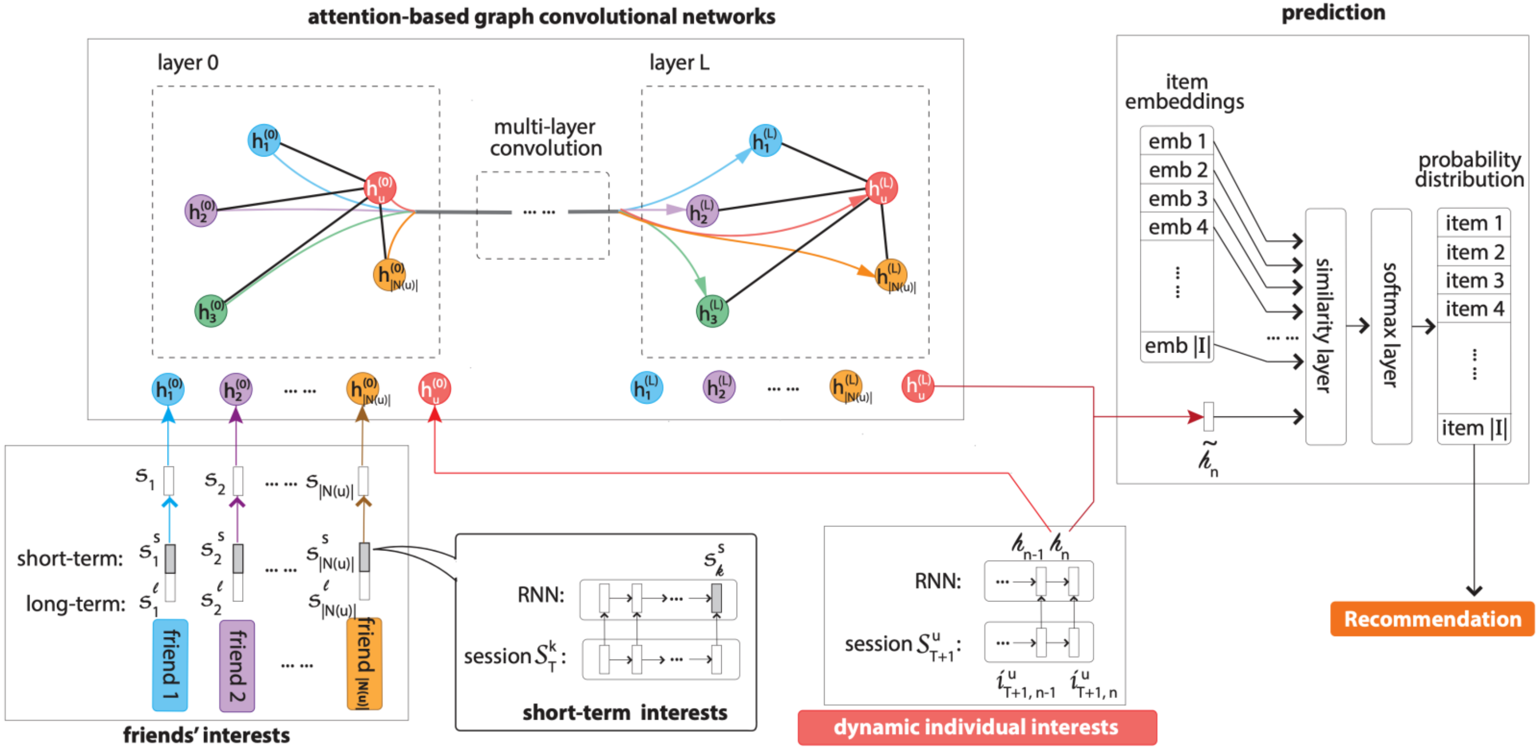
- User information in other domains (such as Social Network) helps to model their preferences.



[Wang X, He X, Nie L, et al. Item silk road: Recommending items from information domains to social users. SIGIR 2017]

1. Using Cross-domain information (2) Dynamic Graph Recommendation

- DGRec uses graph neural network to model preference propagation among friends.



[Song W, Xiao Z, Wang Y, et al. Session-based social recommendation via dynamic graph attention networks. WSDM 2019]

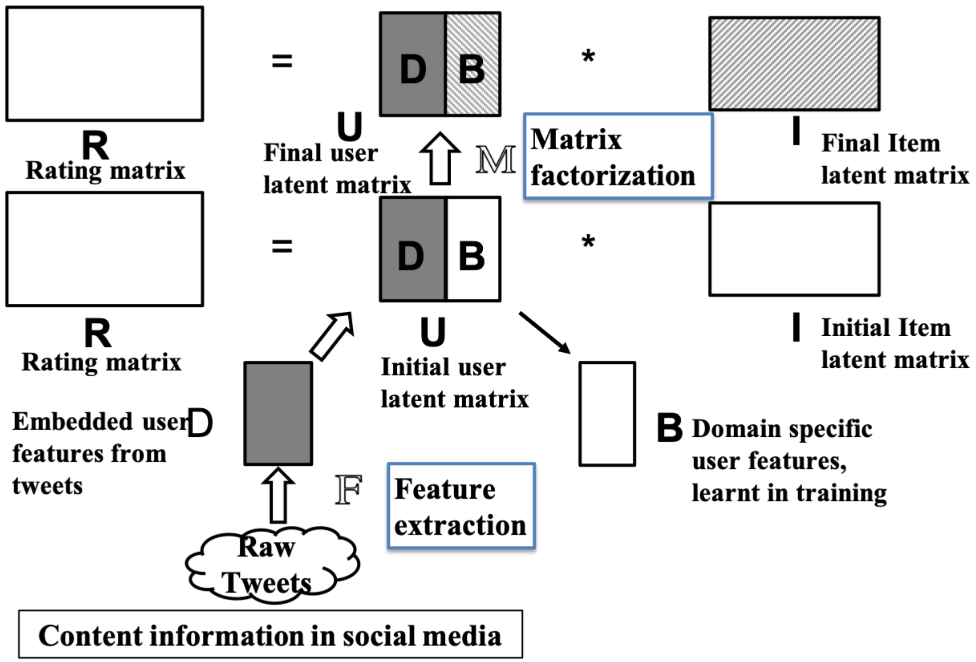
Experimental Results

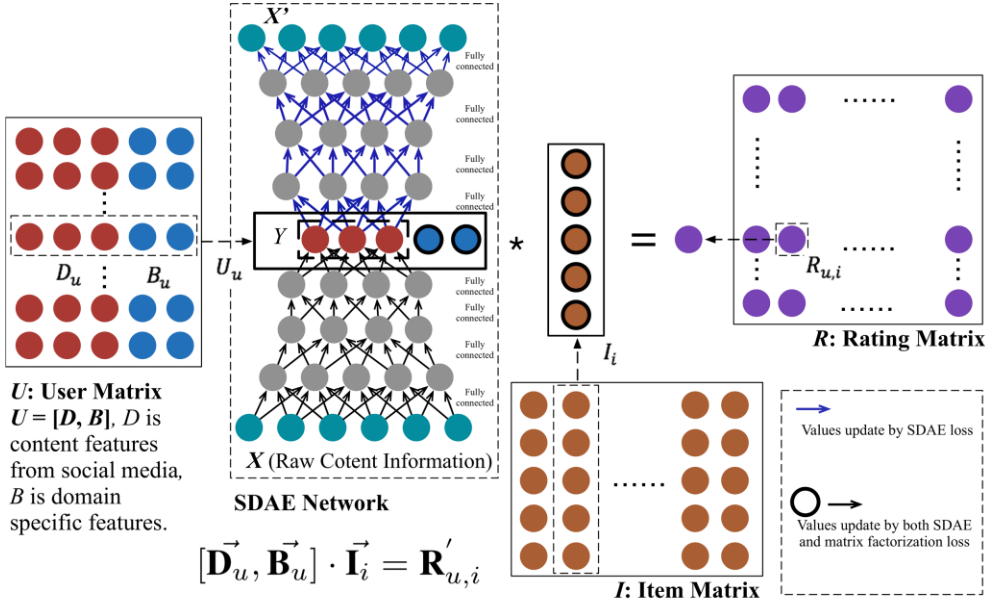
- DGRec combines the social influence and temporal preference in session-based recommendation.

Model Class	Model	Douban		Delicious		Yelp	
		Recall@20	NDCG	Recall@20	NDCG	Recall@20	NDCG
Classical	ItemKNN [22]	0.1431	0.1635	0.2729	0.2241	0.0441	0.0989
	BPR-MF [27]	0.0163	0.1110	0.2775	0.2293	0.0365	0.1190
Social	SoReg [24]	0.0177	0.1113	0.2703	0.2271	0.0398	0.1218
	SBPR [41]	0.0171	0.1059	0.2948	0.2391	0.0417	0.1207
	TranSIV [38]	0.0173	0.1102	0.2588	0.2158	0.0420	0.1187
Temporal	RNN-Session [13]	0.1643	0.1854	0.3445	0.2581	0.0756	0.1378
	NARM [21]	0.1755	0.1872	0.3776	0.2768	0.0765	0.1380
Social + Temporal (Ours)	DGRec	**0.1861**	**0.1950**	**0.4066**	**0.2944**	**0.0842**	**0.1427**

2. Using Cross-media information

- Main idea: cross-media information can be useful too



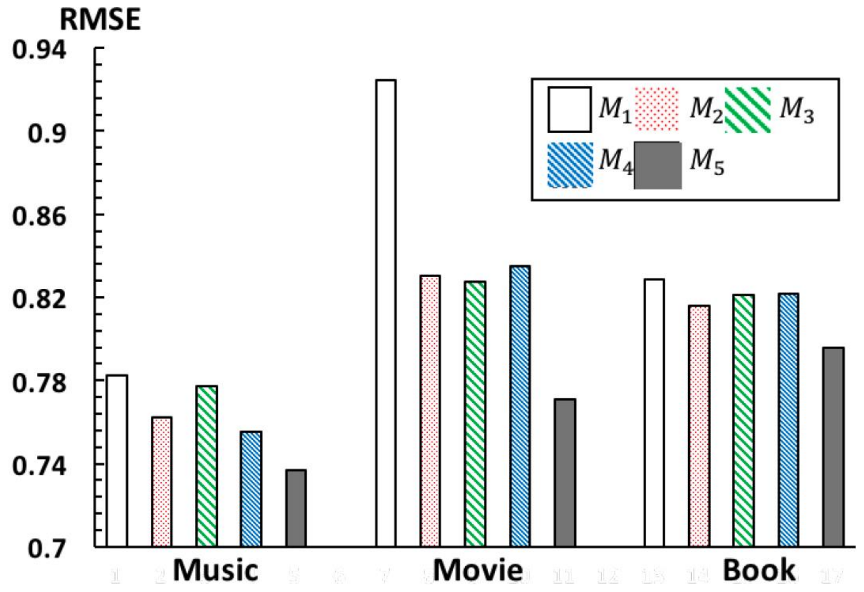


[Ma W, Zhang M, Wang C, et al. Your Tweets Reveal What You Like: Introducing Cross-media Content Information into Multi-domain Recommendation. IJCAI'2018]

Experimental Results

Algorithm		Music	Movie	Book
MF		0.6862	0.7127	0.7698
MF_S	SDAE	**0.6784****	**0.7125**	**0.7595****
	W2V	0.6833**	0.7126	0.7643**
	P2V	0.6789**	0.7125	0.7626**
Greatest Improvement		1.15%	0.03%	1.36%
PMF		0.6881	0.7133	0.7480
PMF_S	SDAE	**0.6766****	0.7102**	0.7391**
	W2V	0.6776**	**0.7101****	**0.7390****
	P2V	0.6776**	0.7102**	**0.7390****
Greatest Improvement		1.70%	0.45%	1.20%
HFT		0.7842	0.7357	0.9632
HFT_S	SDAE	0.7635**	0.7244**	0.9472**
	W2V	0.7639**	0.7257**	0.9341**
	P2V	**0.7567****	**0.7234****	**0.9221****
Greatest Improvement		3.51%	1.67%	4.27%
TrustMF		0.6751	0.7236	0.7438
TrustMF_S	SDAE	**0.6712****	**0.7154****	**0.7372****
	W2V	0.6715**	0.7155**	0.7377**
	P2V	0.6732**	0.7165**	**0.7372****
Greatest Improvement		0.58%	1.15%	0.90%

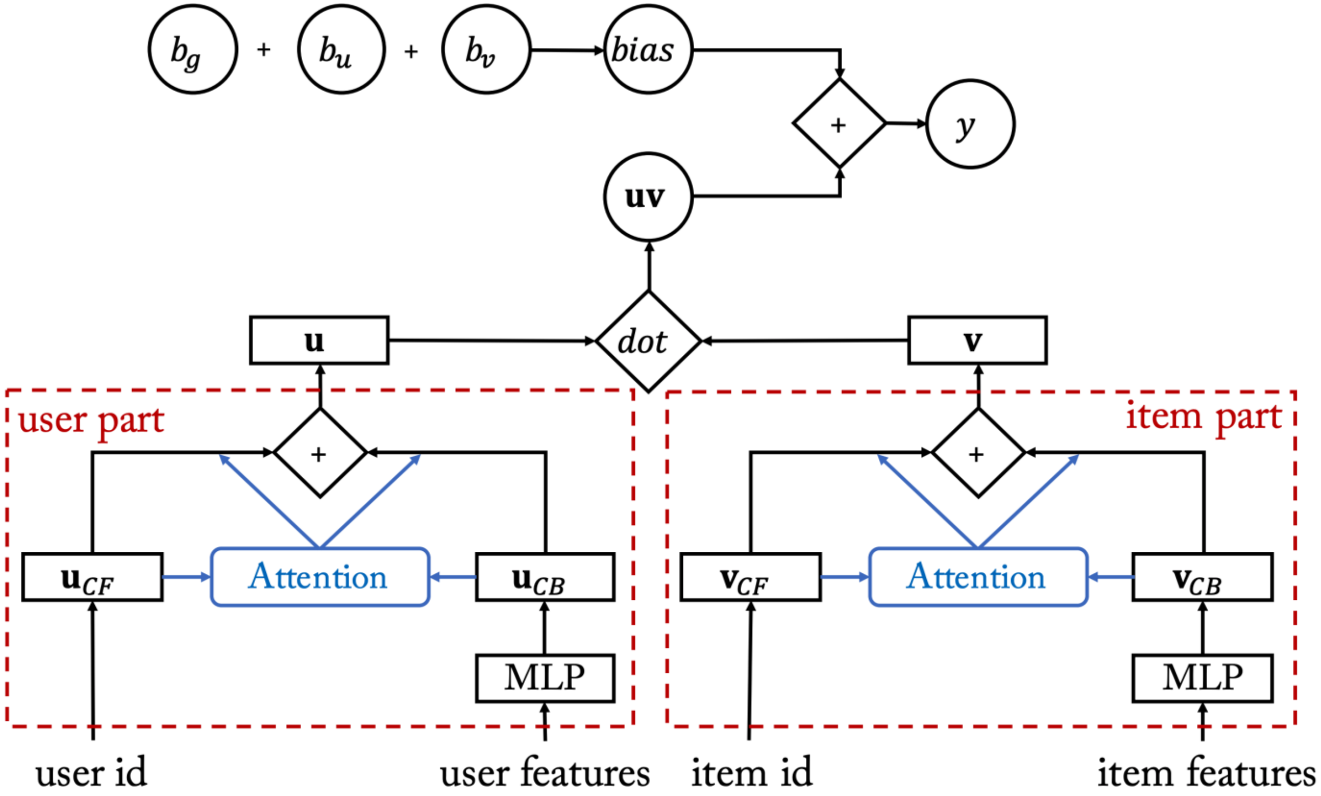
- The incorporation of cross-media (even off-topic) information enables the algorithms to handle new users



The prediction results in new users. The experimental results indicate that M_5 performs significant better than others in each domain ($p < 0.01$)

3. Model level solution: ACCM

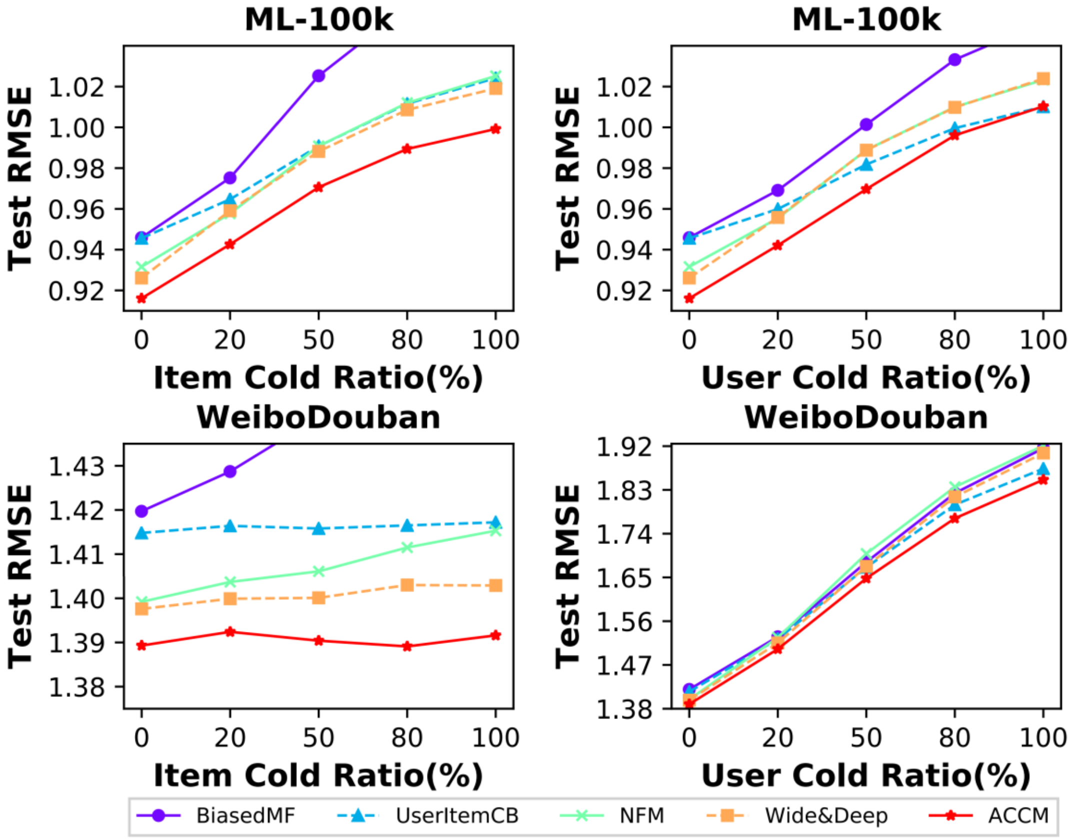
- Attentional Content & Collaborate Model (ACCM)



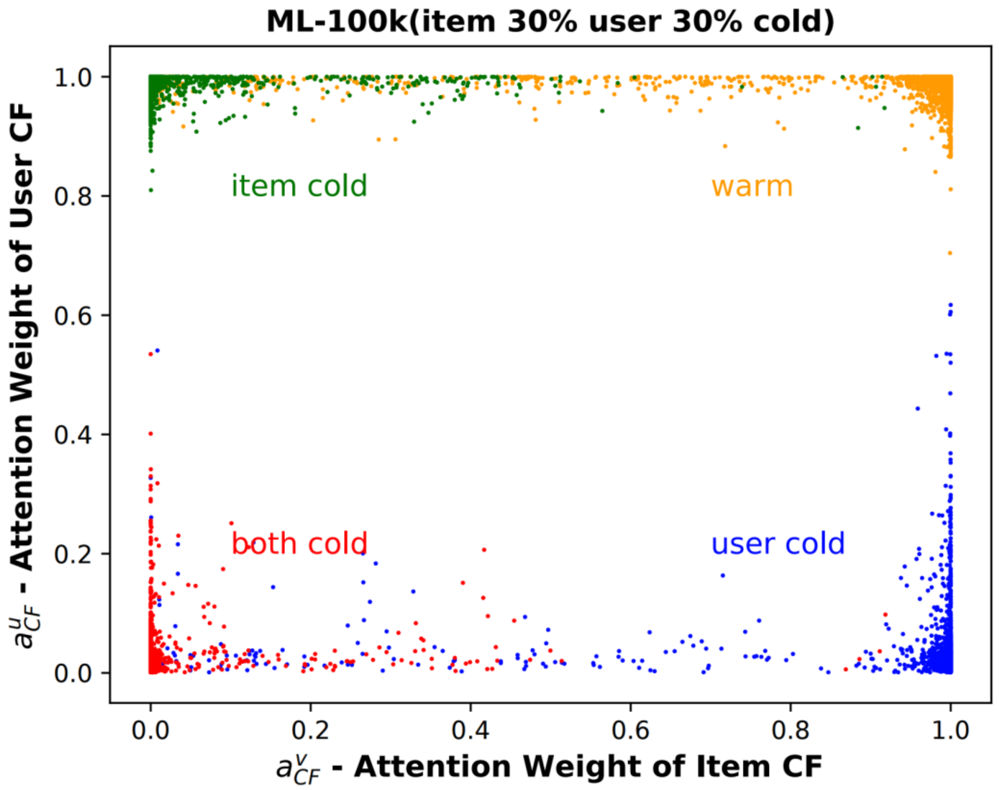
[Shi S, Zhang M, Liu Y, et al. Attention-based adaptive model to unify warm and cold starts recommendation. CIKM 2018]

ACCM – Experimental Results

- Outperform baselines in different scenarios



- Adaptively adjust weights of CF and CB



Trends

- Towards challenges
  - Cold-start
  - **Efficiency**
  - Explainability
  - User satisfaction & behavior
  - Exploration vs. exploitation (Diversity)
  - Fairness
- Trending techniques
  - Knowledge-aware
  - Sequential Recommendation
  - Reinforcement Learning
- New scenarios

Towards Challenges – Efficiency

Information Retrieval @ Tsinghua University

Complex Neural Network

- ✓✓ Exploring new deep learning architectures for Rec. Sys.
  - Attention, Memory Network, Graph NN, GAN, etc
  - Superior ability to complex network structures
- ✓✓ With substantial number of parameters

ed

Can we find some solutions to *efficiently* learn a recommendation model *without sampling*?

Negative Sampling

- ✓✓ Not robust
- ✓✓ Difficult to achieve the optimal performance in practical applications

1. Fast Matrix Factorization with Non-uniform Weights

- An efficient learner by using memoization.
- Key idea: memoizing the computation for missing data part:

$$L(\Theta) = \sum_{(u,i)\in\mathcal{R}} (y_{ui} - \hat{y}_{ui})^2 + \boxed{\sum_{u} \sum_{i \notin \mathcal{R}_u} c_i (0 - \hat{y}_{ui})^2}$$

Bottleneck: Missing data part

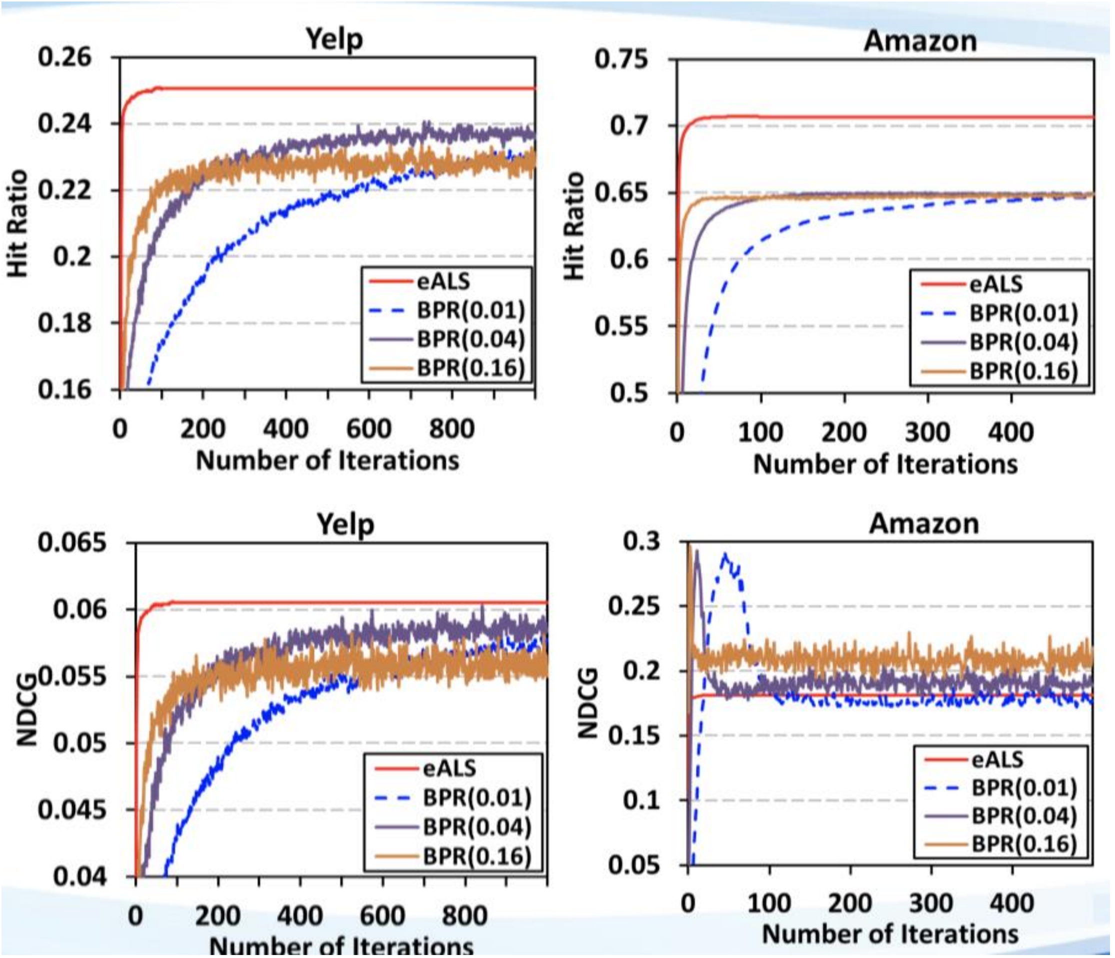
- Reformulating the loss function:

$$L(\Theta) = \sum_{(u,i)\in\mathcal{R}} [(y_{ui} - \hat{y}_{ui})^2 - c_i \hat{y}_{ui}^2] + \boxed{\sum_{u} \sum_{i} c_i \hat{y}_{ui}^2}$$

Sum over all user-item pairs, can be seen as a prior over all interactions!

[XN He, JH Tang, et al, Fast Matrix Factorization for Online Recommendation with Implicit Feedback， SIGIR2016.]

Experimental Results

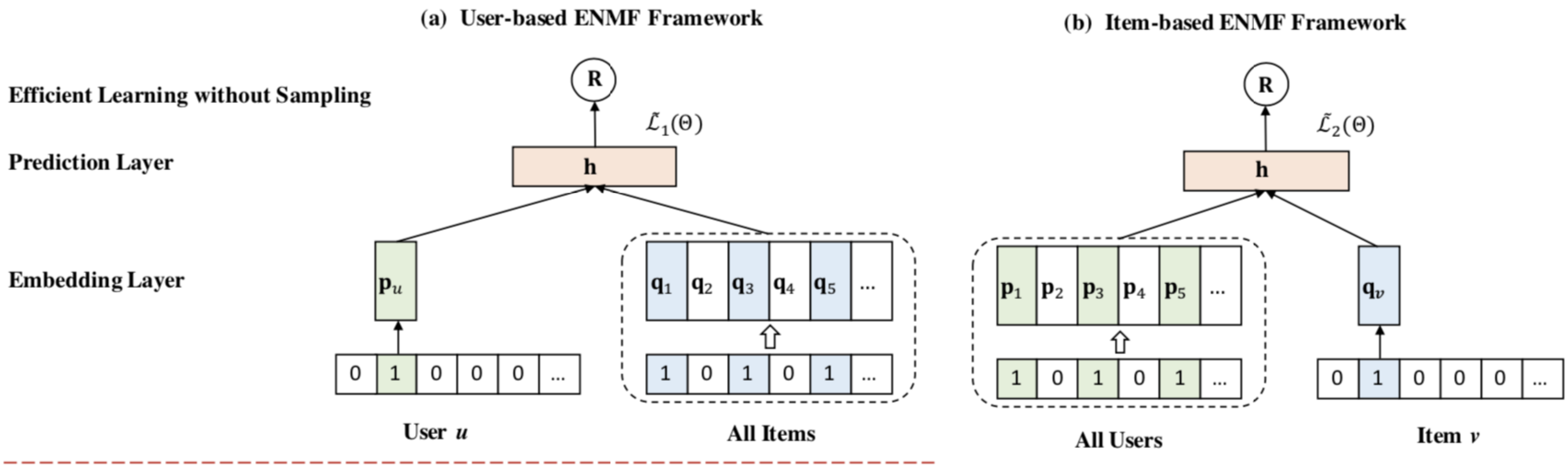


Training time per iteration (Java, single-thread)

	Yelp (0.73M)		Amazon (5M)	
Factor#	**eALS**	**ALS**	**eALS**	**ALS**
32	1 s	10 s	9 s	74 s
64	4 s	46 s	23 s	4.8 m
128	13 s	221 s	72 s	21 m
256	1 m	23 m	4 m	2 h
512	2 m	2.5 h	12 m	11.6 h

69

2. Efficient Non-sampling Neural Model



[C Chen, M Zhang, et al, An Efficient Adaptive Transfer Neural Network for Social-aware Recommendation, SIGIR2019.]

Loss Inference

$$\mathcal{L}_{\mathcal{I}}(\Theta) = \sum_{u\in B}\sum_{v\in V} c^{I}_{uv}(R^2_{uv} - 2R_{uv}\hat{R}_{uv} + \hat{R}^2_{uv})$$

User actions, R_{uv} : (0,1)

R_{uv} =0
for neg. feedbacks,

$$\begin{aligned}
\mathcal{L}_{\mathcal{I}}(\Theta) &= const - 2\sum_{u\in B}\sum_{v\in V^+} c^{I+}_{uv}\hat{R}_{uv} + \sum_{u\in B}\sum_{v\in V} c_{uv}\hat{R}^2_{uv} \\
&= const - 2\sum_{u\in B}\sum_{v\in V^+} c^{I+}_{uv}\hat{R}_{uv} + \sum_{u\in B}\sum_{v\in V^+} c^{I+}_{uv}\hat{R}^2_{uv} + \sum_{u\in B}\sum_{v\in V^-} c^{I-}_{uv}\hat{R}^2_{uv} \\
&= const - 2\sum_{u\in B}\sum_{v\in V^+} c^{I+}_{uv}\hat{R}_{uv} + \sum_{u\in B}\sum_{v\in V^+} c^{I+}_{uv}\hat{R}^2_{uv} + \sum_{u\in B}\sum_{v\in V} c^{I-}_{uv}\hat{R}^2_{uv} - \sum_{u\in B}\sum_{v\in V^+} c^{I-}_{uv}\hat{R}^2_{uv} \\
&= const + \boxed{\underbrace{\sum_{u\in B}\sum_{v\in V} c^{I-}_{uv}\hat{R}^2_{uv}}_{\mathcal{L}^{A}_{\mathcal{I}}(\Theta)}} + \sum_{u\in B}\sum_{v\in V^+}\Big((c^{I+}_{uv} - c^{I-}_{uv})\hat{R}^2_{uv} - 2c^{I+}_{uv}\hat{R}_{uv}\Big)
\end{aligned}$$

Bottleneck

Loss Inference

Complexity: $O(|\mathbf{B}||\mathbf{V}|d)$

$$\mathcal{L}_{\mathcal{I}}(\Theta) = \sum_{u \in B} \sum_{v \in V} c_{uv}^{I} (R_{uv}^2 - 2R_{uv}\hat{R}_{uv} + \hat{R}_{uv}^2)$$

$$\mathcal{L}_{\mathcal{I}}(\Theta) = const + \boxed{\underbrace{\sum_{u \in B} \sum_{v \in V} c_{uv}^{I-} \hat{R}_{uv}^2}_{\mathcal{L}_{\mathcal{I}}^{\mathcal{A}}(\Theta)}} + \sum_{u \in B} \sum_{v \in V+} \left((c_{uv}^{I+} - c_{uv}^{I-}) \hat{R}_{uv}^2 - 2c_{uv}^{I+} \hat{R}_{uv} \right)$$

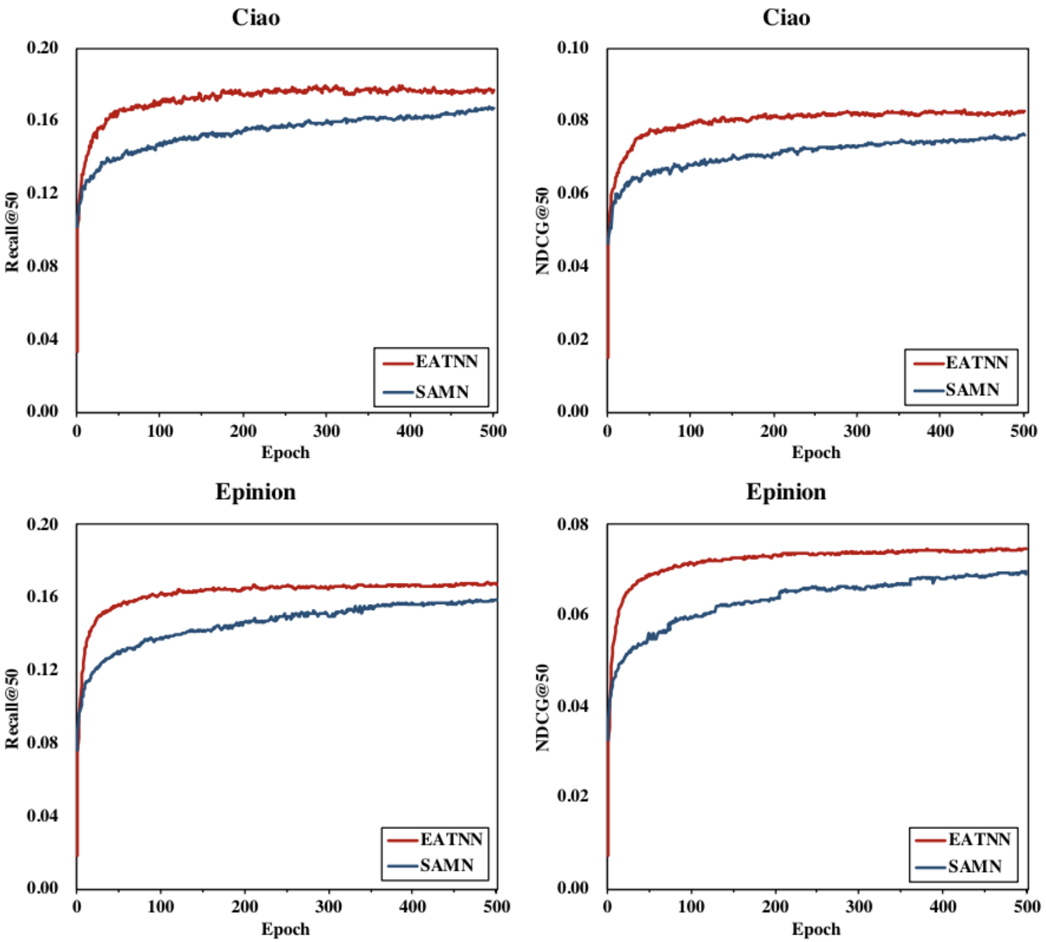
Bottleneck

$$\boxed{\sum_{i=1}^{d} \sum_{j=1}^{d} \left(\underline{\left(\sum_{u \in B} p_{u,i}^{I} p_{u,j}^{I} \right)} \underline{\left(\sum_{v \in V} c_{v}^{I-} q_{v,i} q_{v,j} \right)} (h_{1,i} h_{1,j}) \right)}$$

$$\boxed{\begin{aligned} \hat{R}_{uv}^2 &= \sum_{i=1}^{d} h_{1,i} p_{u,i}^{I} q_{v,i} \sum_{j=1}^{d} h_{1,j} p_{u,j}^{I} q_{v,j} \\ &= \sum_{i=1}^{d} \sum_{j=1}^{d} (p_{u,i}^{I} p_{u,j}^{I}) (q_{v,i} q_{v,j}) (h_{1,i} h_{1,j}) \end{aligned}}$$

Independent, opportunity to speed-up by precomputing the two terms.

Experimental Results



Comparison of runtime

s:second; m: minute; h: hour; d: day

S: training time for a single iteration;

I: Overall iterations;

T: Total time

Model	*Ciao*			*Epinion*			*Flixster*		
	S	I	T	S	I	T	S	I	T
TranSIV	55s	50	46m	410s	50	342m	37m	50	31h
SAMN	31s	500	258m	92s	500	767m	56m	200	8d
EATNN	1.8s	200	6m	11s	200	37m	8m	200	27h

73

Trends

- Towards challenges
  - Cold-start
  - Efficiency
  - **Explainability**
  - User satisfaction & behavior
  - Exploration vs. exploitation (Diversity)
  - Fairness

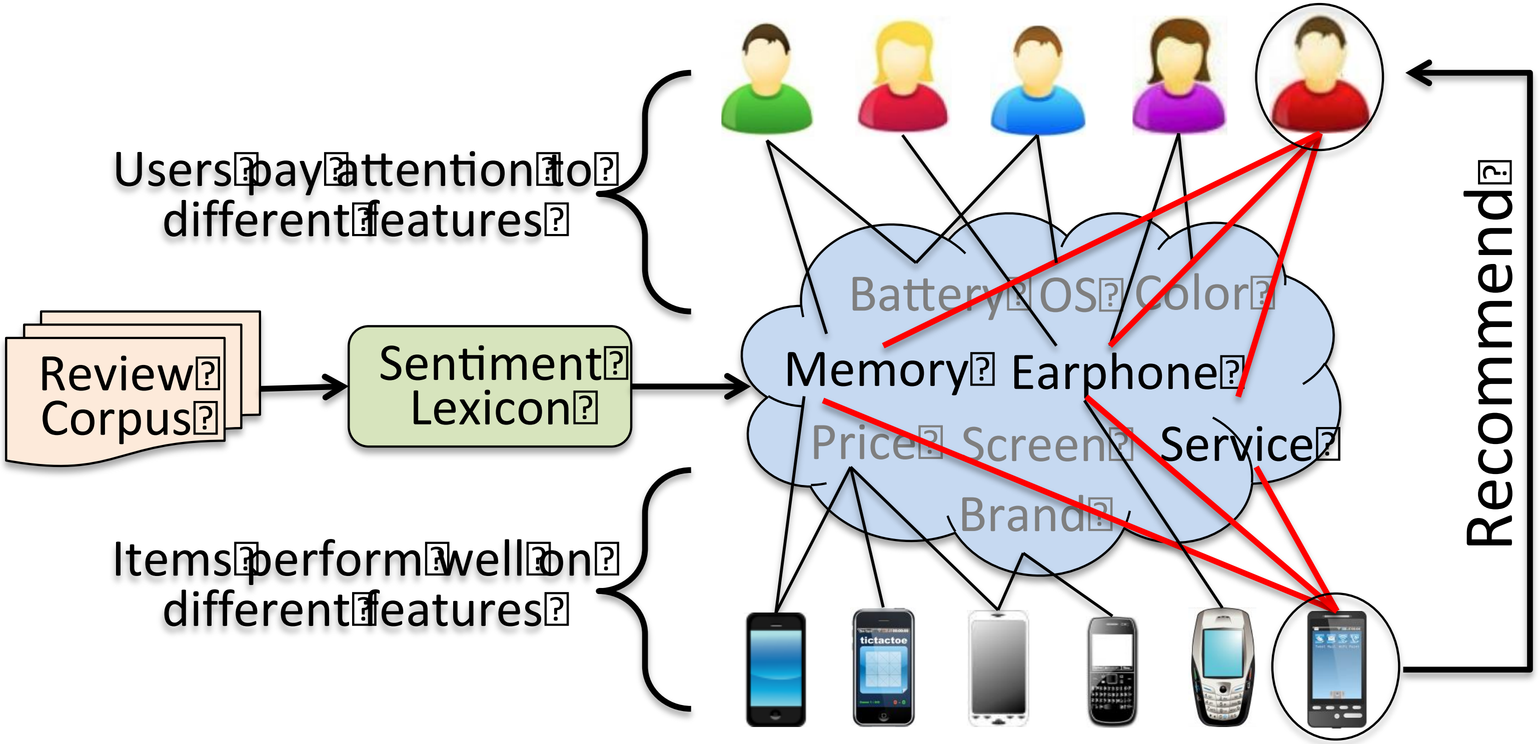
- Trending techniques
  - Knowledge-aware
  - Sequential Recommendation
  - Reinforcement Learning

- New scenarios

Towards Challenges – Explainability

- Aspect-level Explainable Recommendation
- Review-level Explainable Recommendation
- Social-level Explainable Recommendation
- Item-level Explainable Recommendation

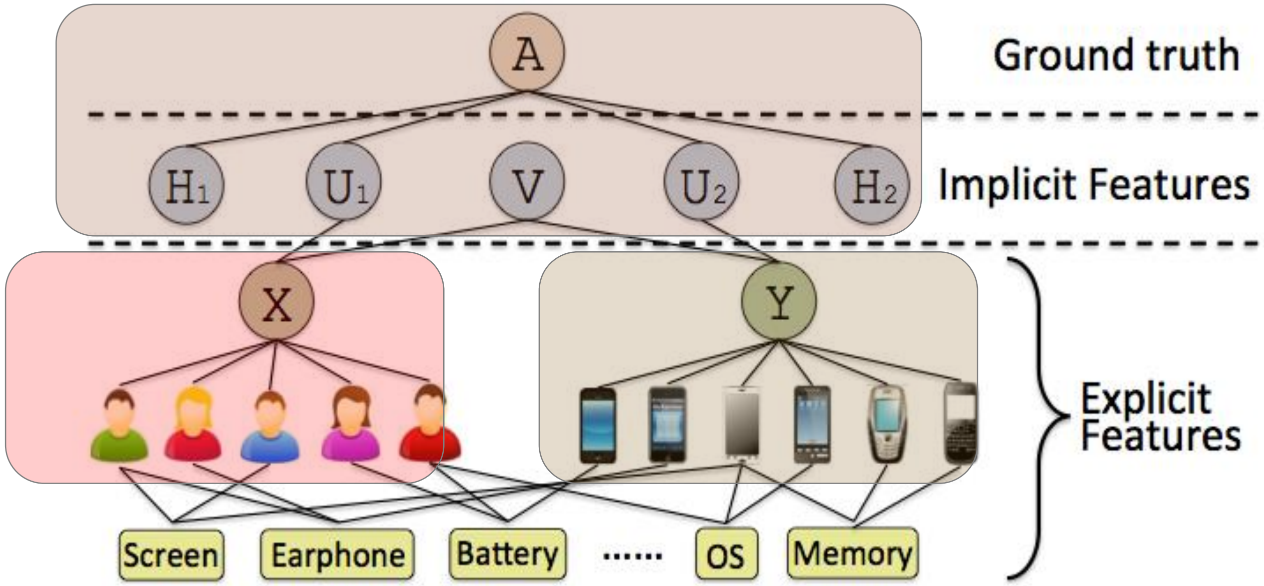
1. Aspect-level explainable recommendation



[YF Zhang, M Zhang, et al, Explicit Factor Models for Explainable Recommendation based on Phrase-level Sentiment Analysis， SIGIR2014.]

[YF Zhang, M Zhang, et al, Rating-Boosted Latent Topics: Understanding Users and Items with Ratings and Reviews, IJCAI 2016]

Framework



$$\underset{U_1,U_2,V,H_1,H_2}{\text{minimize}} \left\{ \|PQ^T - A\|_F^2 + \lambda_x \|U_1 V^T - X\|_F^2 + \lambda_y \|U_2 V^T - Y\|_F^2 + \lambda_u (\|U_1\|_F^2 + \|U_2\|_F^2) + \lambda_h (\|H_1\|_F^2 + \|H_2\|_F^2) + \lambda_v \|V\|_F^2 \right\}$$

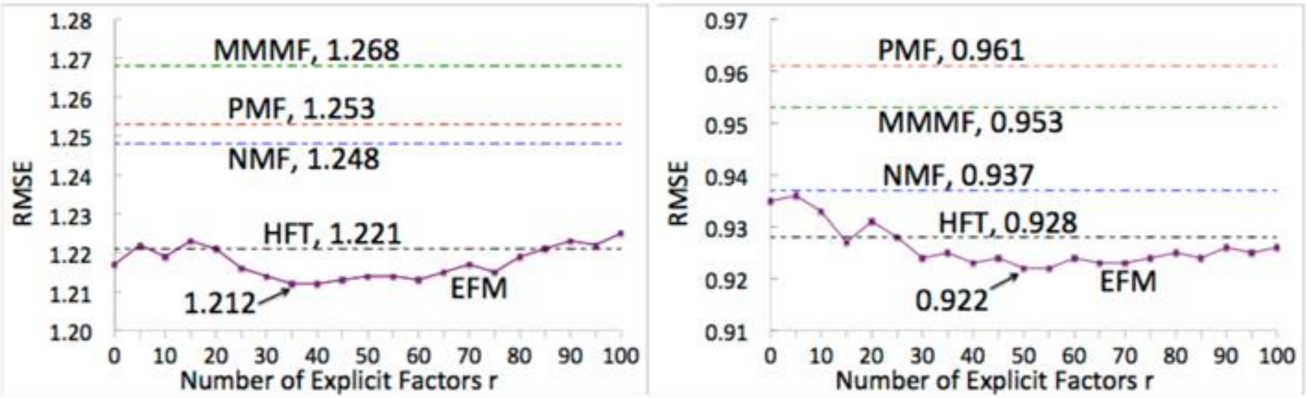
$$P = [U_1 \; H_1], \; Q = [U_2 \; H_2]$$

Explicit Factors

Hidden Factors

Results: explanations do help

- Offline test on public available datasets: Dianping, Yelp



(a) Yelp dataset (b) Dianping dataset

- Online test on E-commerce website

User Set	A		B		C	
Records	#Record	#Click	#Record	#Click	#Record	#Click
	15,933	691	11,483	370	17,265	552
CTR	4.34%		3.22%		3.20%	

2. Review-level explainable recommendation

A

An Awesome Movie! (5 stars)

By Jokerz Wild on October 9, 2017

Format: Amazon Video | Verified Purchase

I love Iron Man!

B

Comic book characters... making millions of horrible movies these days. (1 star)

By TylerVogt3329 on November 14, 2008

Format: DVD

You people these days consider this a good movie? Haha. Who in their right mind believes that a rich playboy can save the world from evil?
For good and REAL action check out WWE, ECW, or TNA.
For good classic Wrestling.. check out WCW and WWF.

C

Rating (to the item)

Good solid film (4 stars)

By M-M on July 30, 2013

Format: Amazon Video | Verified Purchase

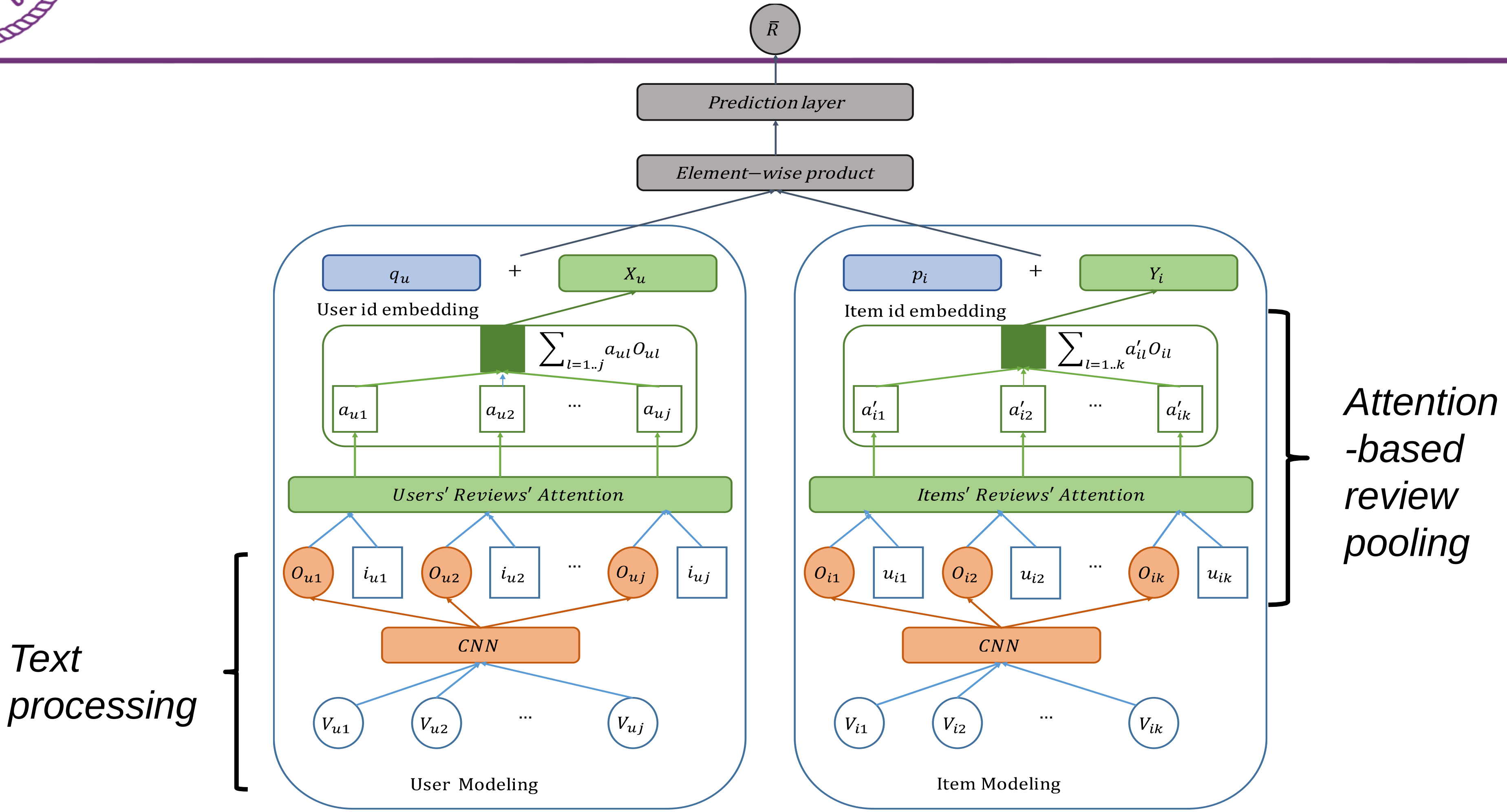
Review (to the item)

It turned out to be entertaining and at the end I enjoyed the film. Good special effects, nice story line for "actions" and "comics". The protagonist (Tony Stark) looks natural: arrogant, brash, but at the same time clever, intelligent and ethic. The villain is a little bit overreacting, and annoying, as most of antagonists :) Overall that's a good movie.

Rated usefulness (to the review)

8 people found this helpful. Was this review helpful to you? Yes No Report abuse

Framework



[C Chen, M Zhang, et al, Neural Attentional Rating Regression with Review-level Explanations, WWW 2018]

Results: review usefulness

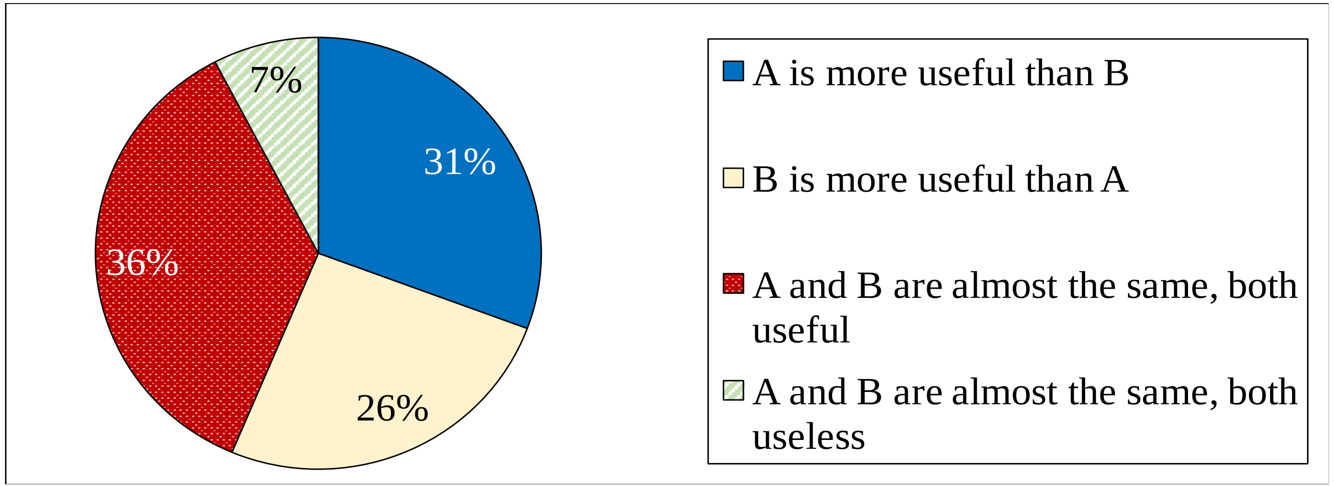
- Ground truth: Top_rated_useful   Latest， Length：Bias / unfairness

	Toys_and_Games				Kindle_Store				Movies_and_TV			
	Latest	Random	Length	NARRE	Latest	Random	Length	NARRE	Latest	Random	Length	NARRE
Precision@1	0.1487	0.3255	0.2476	**0.3860****	0.2447	0.4574	0.4041	**0.5235****	0.3040	0.4908	0.3903	**0.6576****
Recall@1	0.0362	0.0952	0.0771	**0.1398****	0.0400	0.0992	0.0852	**0.1131****	0.0436	0.0976	0.0677	**0.1445****
Precision@10	0.1550	0.2000	0.2316	**0.2697****	0.2228	0.2707	0.2933	**0.3530****	0.2325	0.2925	0.3369	**0.3459****
Recall@10	0.4367	0.5763	0.6763	**0.8601****	0.4510	0.5551	0.6168	**0.8317****	0.3716	0.4673	0.5403	**0.7674****

Bias/unfairness in users' votes: The Matthew Effect

- Ground truth:
  - Crowdsourcing

A: Ours;
B: Top_rated_useful;

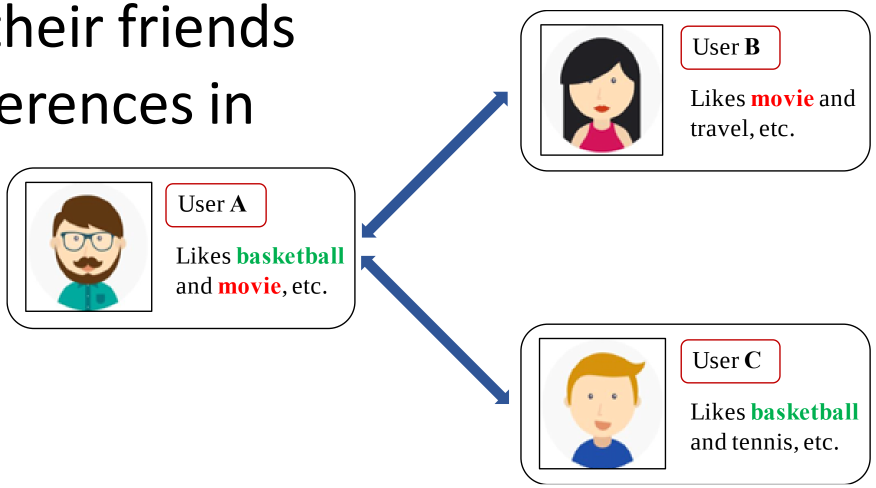


	Precision@1	Precision@5	Precision@10	Recall@1	Recall@5	Recall@10	NDCG@1	NDCG@5	NDCG@10
Top_Rated_Useful	0.4800	0.4440	0.3610	0.0821	0.3453	0.4953	0.6640	0.6906	0.7076
NARRE	**0.5900****	**0.4760****	**0.3850****	**0.1067****	**0.3532****	**0.5046****	**0.7413****	**0.7231****	**0.7358****

3. Social-level Explainable Recommendation

Aspect-level differences

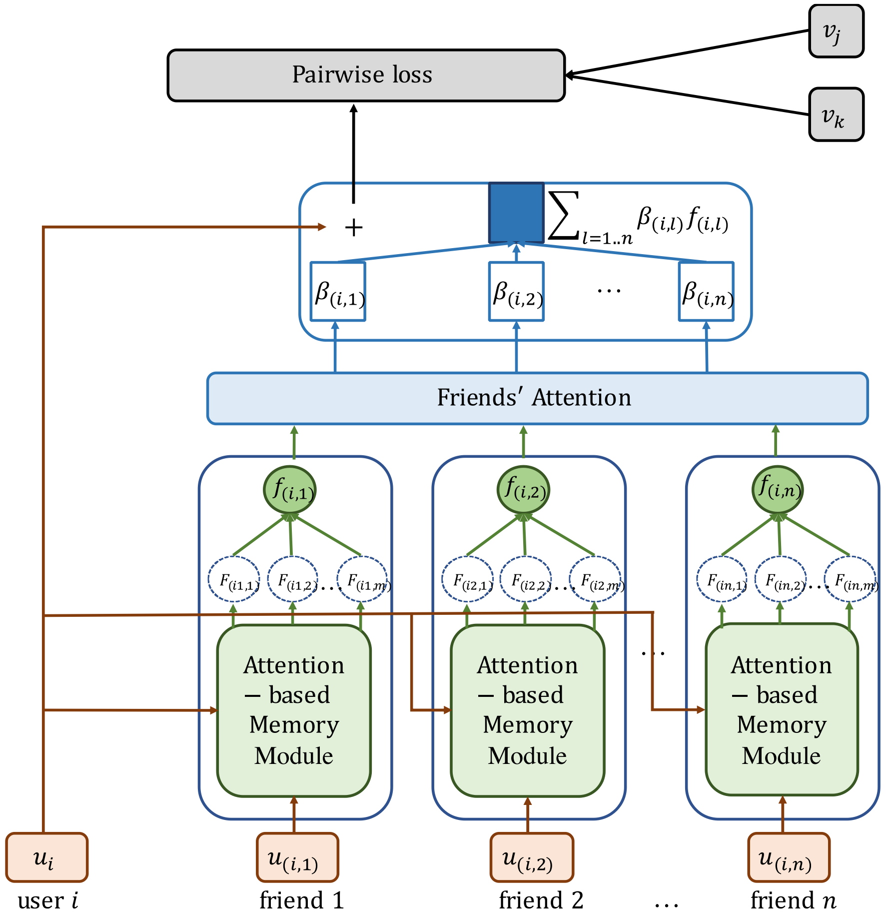
✓✓ Generally, users and their friends only have the same preferences in certain aspects

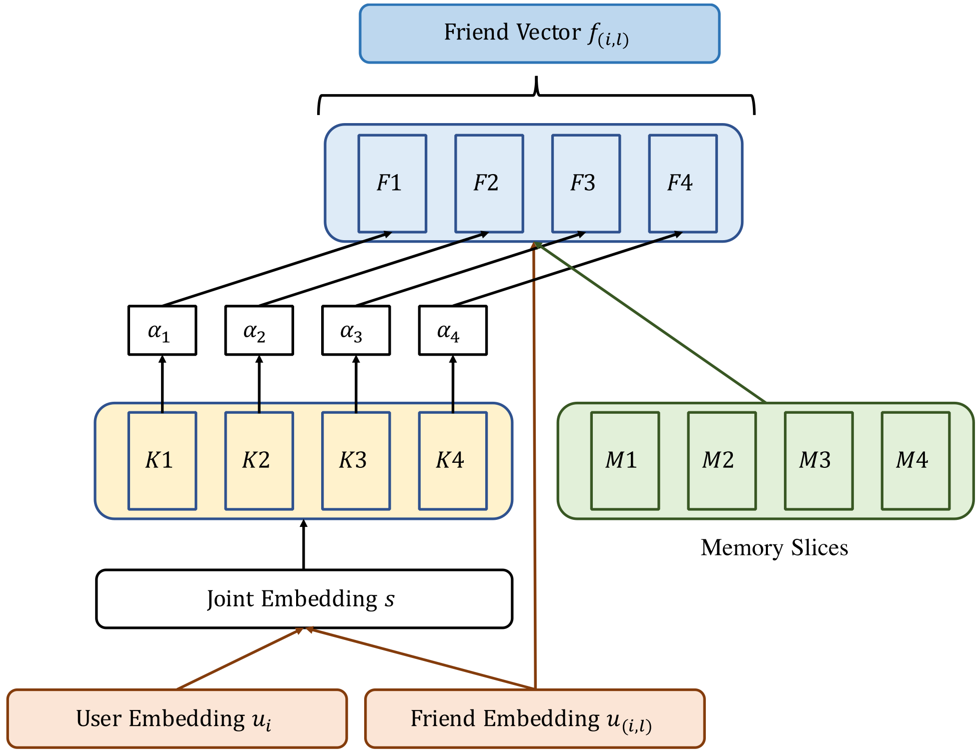


Friend-level differences

✓✓ The influence strength of a user's friends should be dynamically changed when facing different scenarios

Proposed Framework: SAMN





[C Chen, M Zhang, et al, Social Attentional Memory Network for Personalized Recommendation， WSDM 2019]

Experimental Results

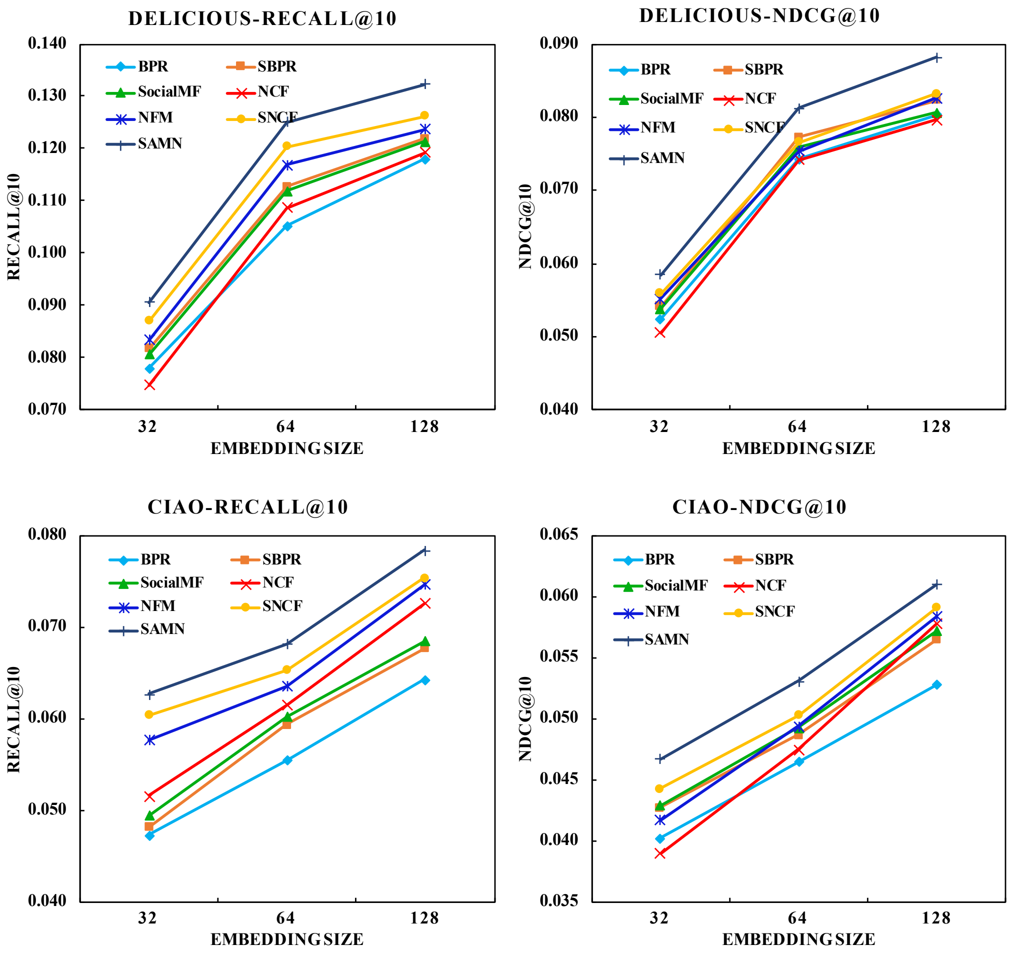
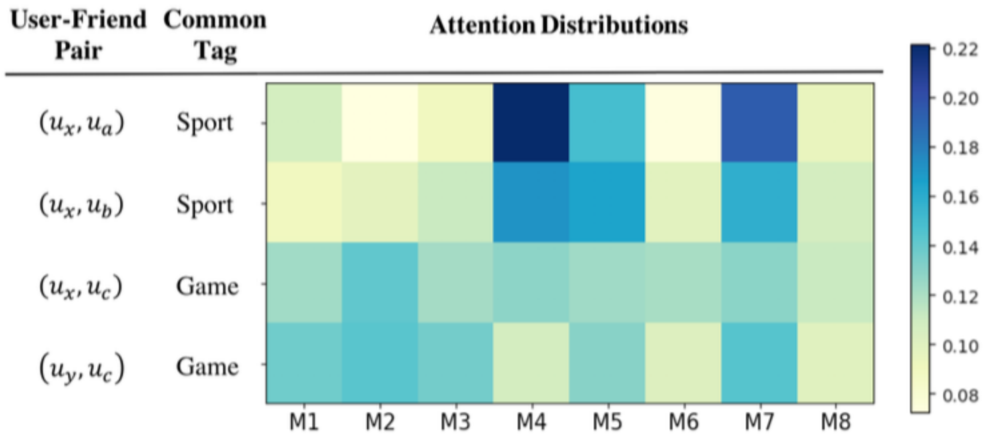
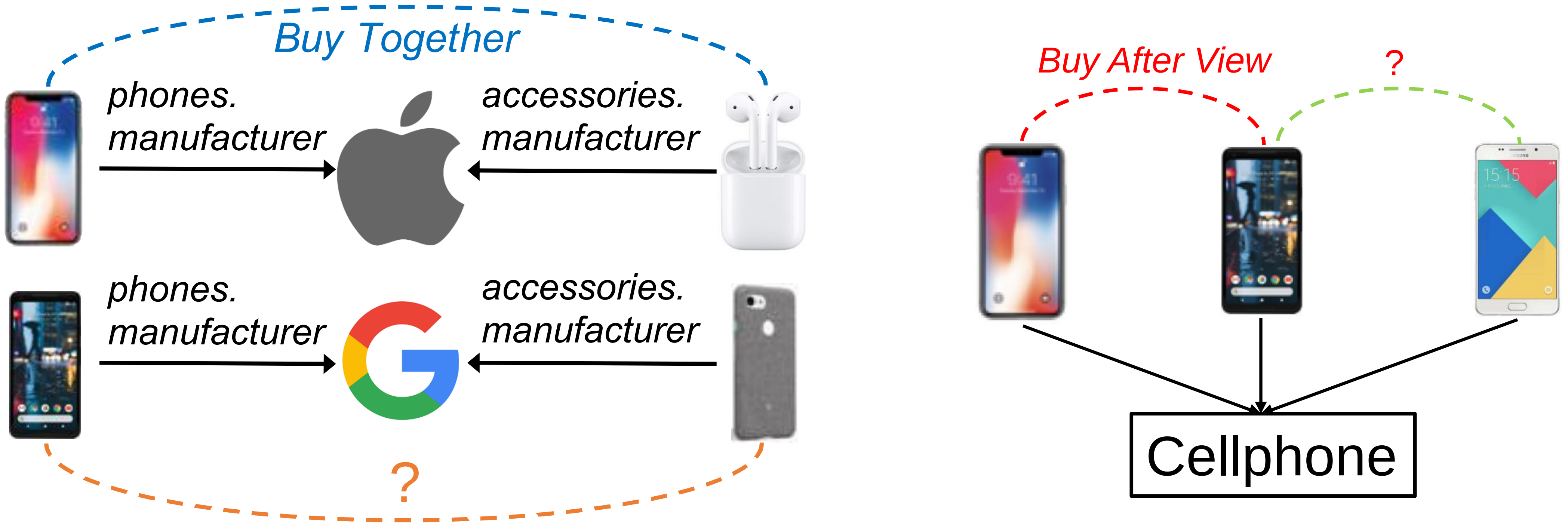


Table 4: Case studies of friend-level attention of a sampled user from Ciao. The friend weights of the user for positive items (Item #130, #212, #1258) and negative items (Item #29, #1105, #3367) are shown.

	Friend #782	Friend #1391	Friend #1446	Friend #1505
Item #130	0.212	0.057	0.319	**0.412**
Item #212	0.145	0.086	**0.517**	0.252
Item #1258	**0.533**	0.121	0.079	0.267
Item #29	0.286	0.103	**0.315**	0.296
Item #1105	**0.434**	0.187	0.315	0.064
Item #3367	0.308	0.075	**0.382**	0.235

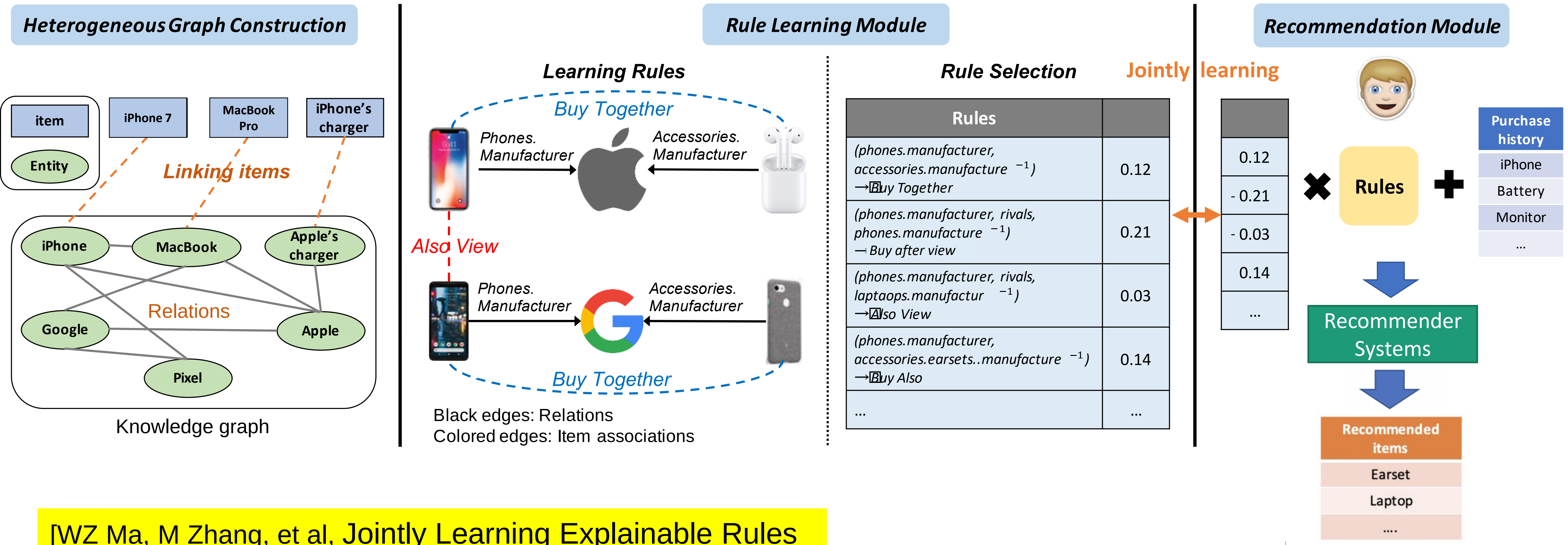


4. RuleRec: KG for Item Recommendation



Black edges: Relations
Colored edges: Item associations

Proposed Framework: RuleRec



[WZ Ma, M Zhang, et al, Jointly Learning Explainable Rules for Recommendation with Knowledge Graph, WWW'2019.]

Experimental Results

Table 5: Performance Comparison between RuleRec and Other Methods in Different Domains. RuleRec_{two} and RuleRec_{multi} are our proposed models. RuleRec_{two} is a two-step rule-based model and RuleRec_{multi} is a multi-task model. These models use BPRMF or NCF as a recommendation model. * indicates statistical significance at $p < 0.01$ compared to the best baseline model.

	Methods / Dataset	Cellphone				Electronic			
		Recall@5	Recall@10	NDCG@10	MRR@10	Recall@5	Recall@10	NDCG@10	MRR@10
Classical baselines	**BPRMF** [27]	0.3238	0.4491	0.2639	0.2058	0.1886	0.2763	0.1571	0.1207
	GMF [13]	0.3379	0.4666	0.2789	0.2223	0.1988	0.2835	0.1657	0.1298
	MLP [5]	0.3374	0.4779	0.2790	0.2182	0.2000	0.2883	0.1681	0.1315
	NCF [14]	0.3388	0.4751	0.2761	0.2151	0.2005	0.2916	0.1679	0.1300
KG+Rec baselines	$\textbf{Hec}_{sl}$ [31]	0.2436	0.3481	0.2040	0.1600	0.1870	0.2851	0.1534	0.1135
	$\textbf{Hec}_{pl}$ [31]	0.2511	0.3564	0.2090	0.1641	0.1948	0.2851	0.1628	0.1256
	RippleNet [34]	0.2834	0.4042	0.2219	0.1780	0.1965	0.2865	0.1638	0.1265
Proposed	$\textbf{RuleRec}_{two}$ (BPRMF)	0.3495*	0.4768	0.2813*	0.2201*	0.2050*	0.2932	0.1707*	0.1334*
	$\textbf{RuleRec}_{multi}$ (BPRMF)	0.3568*	0.4829*	0.2864*	0.2246*	0.2071*	0.2946*	**0.1718***	**0.1341***
	$\textbf{RuleRec}_{two}$ (NCF)	0.3538*	0.4876*	**0.2902***	**0.2296***	0.2049*	**0.2947***	0.1681	0.1296
	$\textbf{RuleRec}_{multi}$ (NCF)	**0.3569***	**0.4894***	**0.2902***	0.2290*	**0.2074***	0.2917	0.1702*	0.1330

Trends

- Towards challenges
  - Cold-start
  - Efficiency
  - Explainability
  - **User satisfaction & behavior**
  - Exploration vs. exploitation (Diversity)
  - Fairness
- Trending techniques
  - Knowledge-aware
  - Sequential Recommendation
  - Reinforcement Learning
- New scenarios

Towards Challenges – Satisfaction&Behavior

Information Retrieval @ Tsinghua University

- Aims of recommender system:
  - Find items match user preference
  - Improve user satisfaction
- Challenge:
  - Two factors are all ***subjective*** and ***implicit***

Entertainment RSs

News RSs

From accuracy to satisfaction

- Most of researchers are:
  - Improving the accuracy of RSs
    - Prediction accuracy, Top-N accuracy
- Most of companies are:
  - Optimizing user behavior signals:
    - CTR, Purchase, …

➔ Final target: user satisfaction

➔ Learn from user behavior

Entertainment RSs

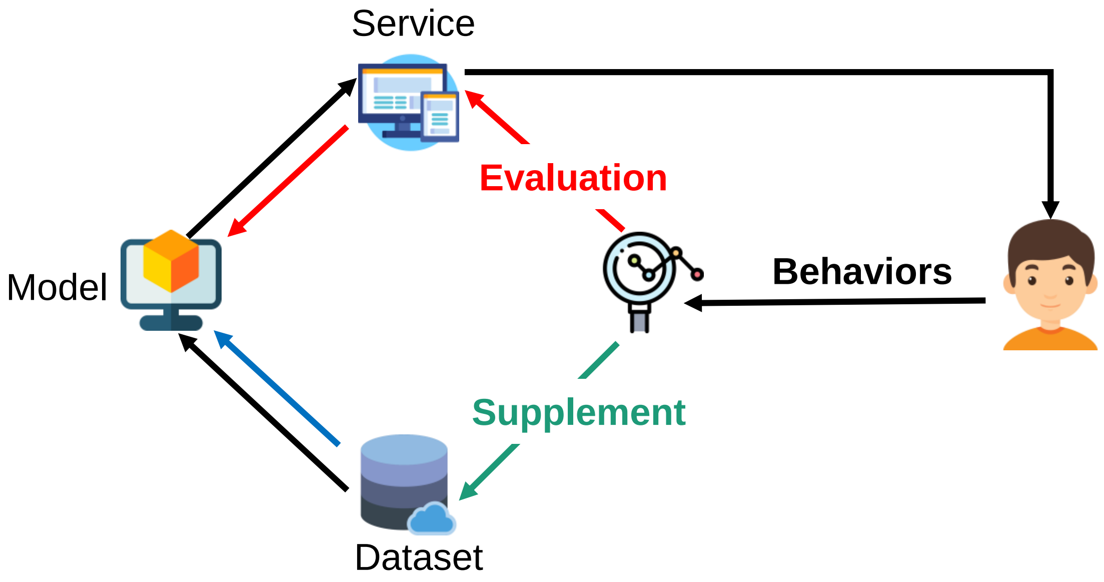
Spotify · iTunes · MUSIC · Google play · DEEZER · TIDAL · SHAZAM · amazon music · anghami · beatport PRO · 7digital · Rhapsody · emusic · junodownload · 247

News RSs

Learn from user behavior

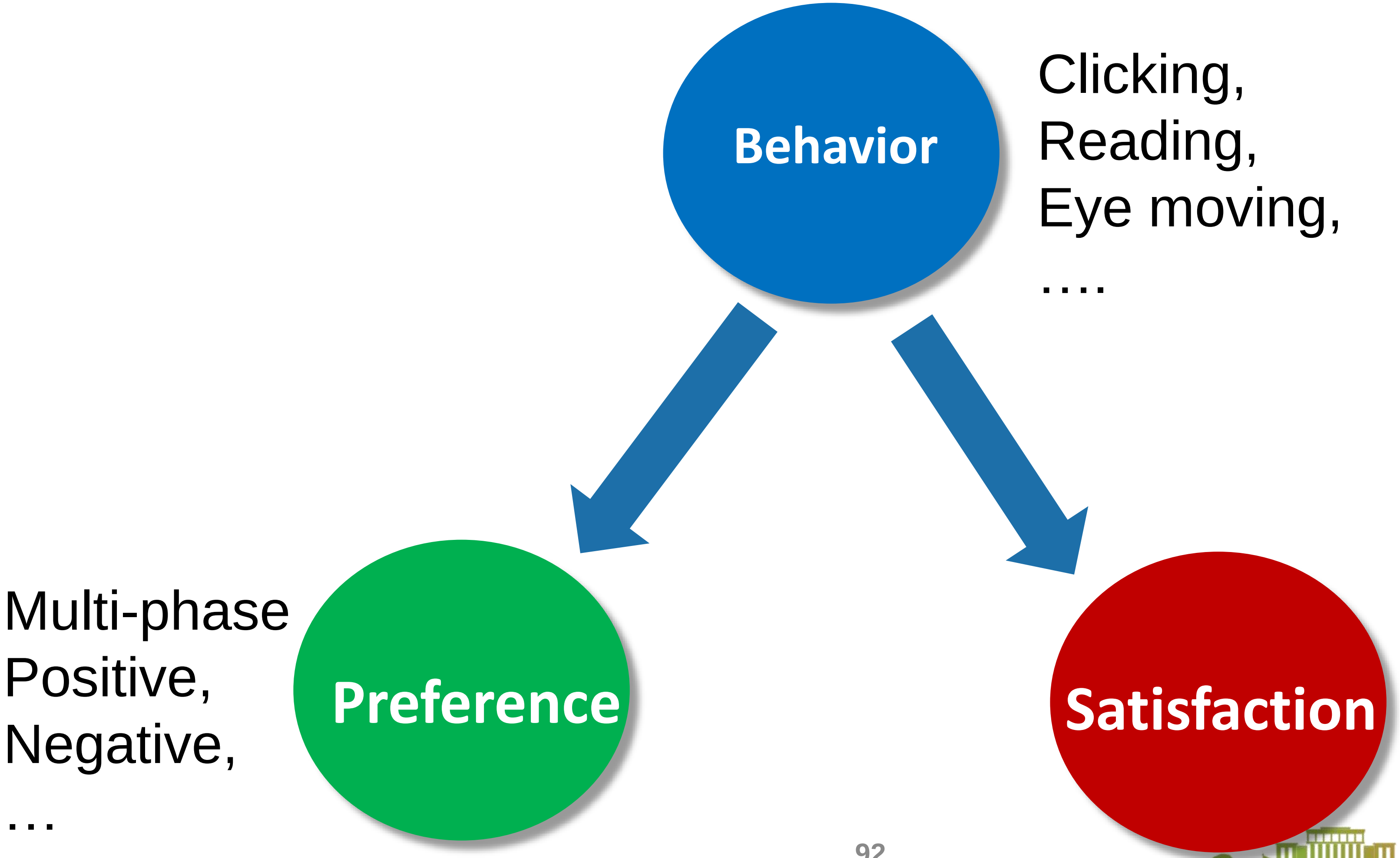
- **How to learn user preference?**
  - From user historical behaviors

    User-item interaction: Explicit, Implicit
- **How to evaluate the performance?**
  - Online: Click-Through-Rate, …
  - Offline: Precision, Recall, nDCG, …



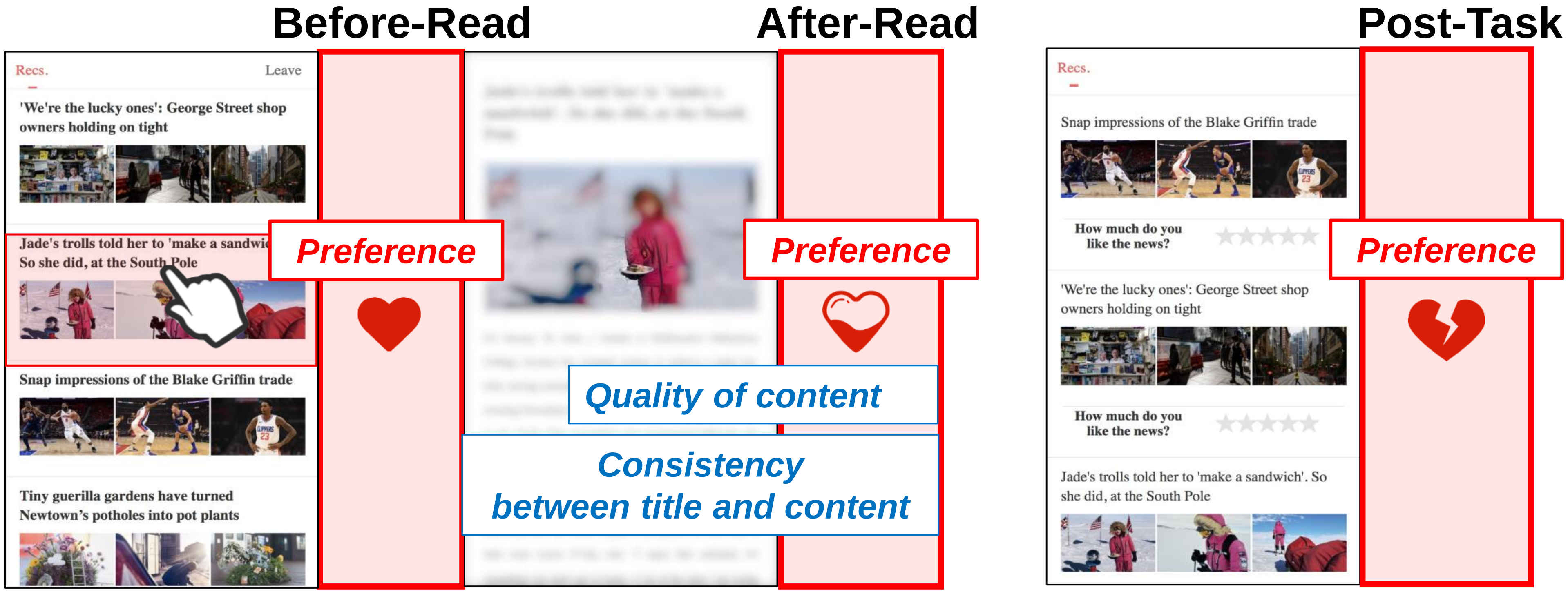
➔ Model relationships between user behaviors and user satisfaction

Behavior, Preference, Satisfaction



1. User click vs. satisfaction

Methodology: multi-phase preferences



Q: How do you expect to prefer reading this piece of news? (5-point Likert scale)

Q: How do you like reading this piece of news? (5-point Likert scale)

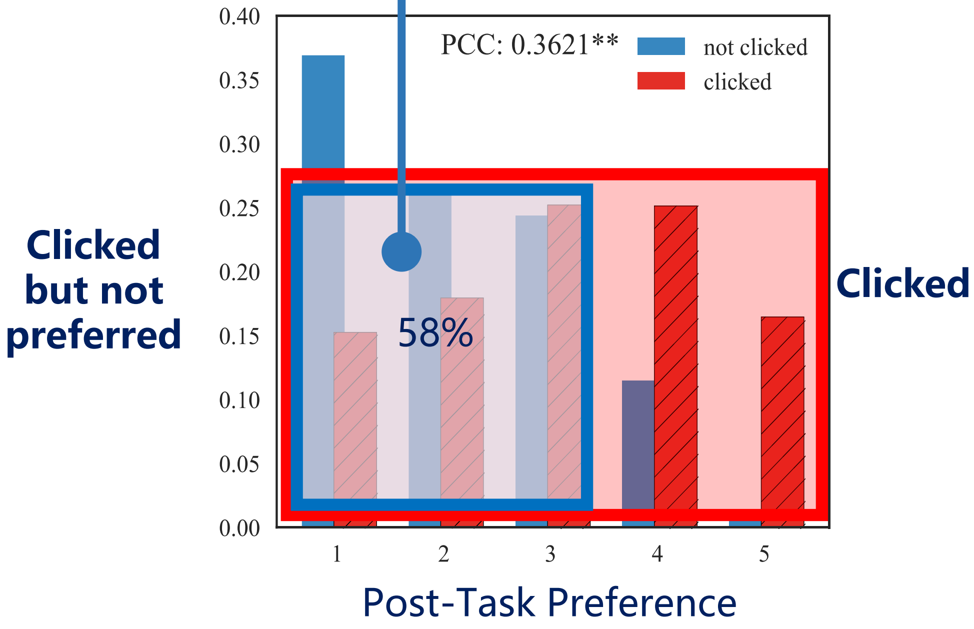
Q: How do you like reading this piece of news? (5-point Likert scale)

[HY Lu, M Zhang, et al. Between Clicks and Satisfaction: Study on Multi-Phase User Preferences and Satisfaction for Online News Reading. In SIGIR ’18]

Result: Bridge click and satisfaction

Do clicks represent user actual preference?

More than half clicked news is not preferred by the user

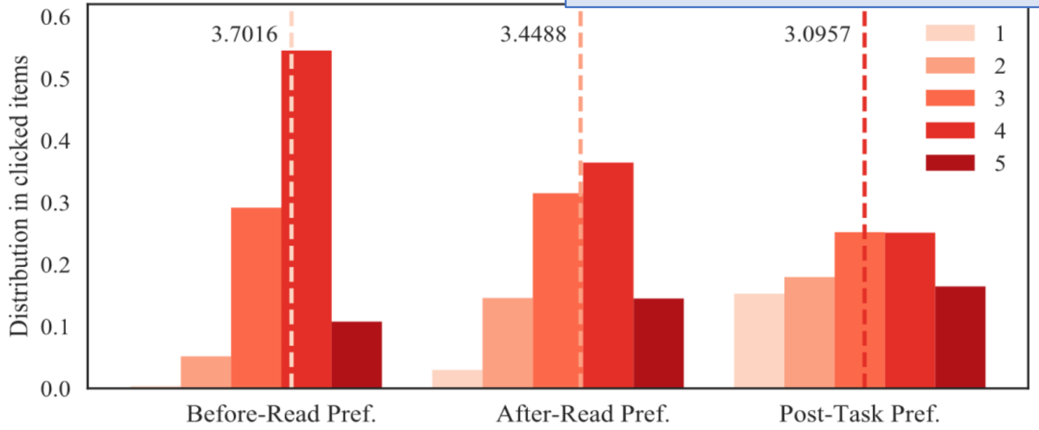


Three Gaps & Reasons Behind

GAP1: Click → Before-Read Pref.

GAP2: Before-Read Pref. → After-Read Pref.

Gap3: After-Read Pref. → Post –Task Pref.



- Gap (1/3): Topic preference & Quality
- Gap (2/3): Quality
- Gap (3/3): Context influence

Result: Application

Information Retrieval @ Tsinghua University

Can we predict user actual preference?

Table 6: Features to predict item-level preference

Behavior features F_b	
B1	Viewport time
B2-B3	Dwell time; Normalized dwell time (in user)
B4-B5	Read -length; -ratio
B6	Read speed
B7	Max scroll interval
B8	Direction change times
Context features F_c	
C1-C4	Dwell time; Read-le[illegible]
C5-C8	Average dwell time [illegible] previous clicks
EQ	Expert labeled quality
Quality features F_q	
Q1	Image num
Q2-Q3	Content / title length
Q4	Stopword num in title
Q5	Similarity of title and content

User behaviors

Interaction context

News quality

Table 7: Results for Liked-Click Prediction (classification).

	Precision	Recall	F-measure	Accuracy
Bin-Click	0.4248	1.0000	0.5963	0.4248
Sat-Click	0.5000	0.3008	0.3757	0.5751
① F_b	0.5570	0.4782	0.5133	0.6164
② F_b+F_c	0.5705**	0.4676	0.5128	0.6239**
③ $F_b+F_c+F_q$	0.5733	0.5007**	0.5338**	0.6293*
④ F_b+F_c+EQ	**0.5818****	**0.5111****	**0.5431****	**0.6358****

The difference between ②&①, ③&②, ④&② are tested by t-test (*means p-value<0.05, **means p-value<0.01).

Table 8: Results for *Post-Task Pref.* Prediction(regression)

	MSE	MAE	PCC
Sat-Click	-	-	0.1400
① F_b	1.2116	0.9099	0.2873
② F_b+F_c	1.1683**	0.8914**	0.3291**
③ $F_b+F_c+F_q$	1.1587*	0.8890	0.3475**
④ F_b+F_c+EQ	**1.1331****	**0.8789****	**0.3548****

Improve online evaluation metrics

Sessi on

$$\text{CTR} = \frac{\sum Clicked(i)}{\#imp} \quad \frac{\sum Gain(i)}{\#imp} \quad \frac{\sum Pref(i)}{\#imp} \quad \frac{\sum Predicted\_user\_pref(i)}{\#imp}$$

Metri

Table 9: Correlation with list-level satisfaction *L-SAT*. (*means p-value<0.05, **means p-value<0.01)

	Bin-click	Sat-Click	Predicted pref.
CG/#imps	0.3765**	0.2547	**0.4521****
CG/#clicks	-	-0.0601	**0.3503****
CG/pos_{lc}	0.2784*	0.1625	**0.3856****
DCG	0.2965*	0.2939*	**0.4134****

Table 10: Concordance with list-level pair-wise satisfaction.(*means p-value<0.05, **means p-value<0.01)

	Bin-click	Sat-Click	Predicted pref.
CG/#imps	0.5667**	0.4000	**0.7333****
CG/#clicks	-	0.4833	**0.6500***
CG/pos_{lc}	0.6500*	0.5000	**0.7333****
DCG	0.6167*	0.5000	**0.6667****

Metrics based on **predicted preference** correlate better with SAT than **binary click signals** do.

[HY Lu, M Zhang, et al. Between Clicks and Satisfaction: Study on Multi-Phase User Preferences and Satisfaction for Online News Reading. In SIGIR '18]

2. Dwell time vs. preference

Dwell time is related with preference

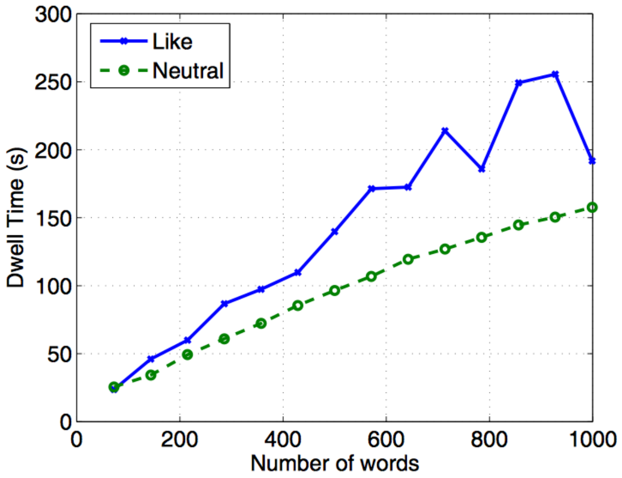
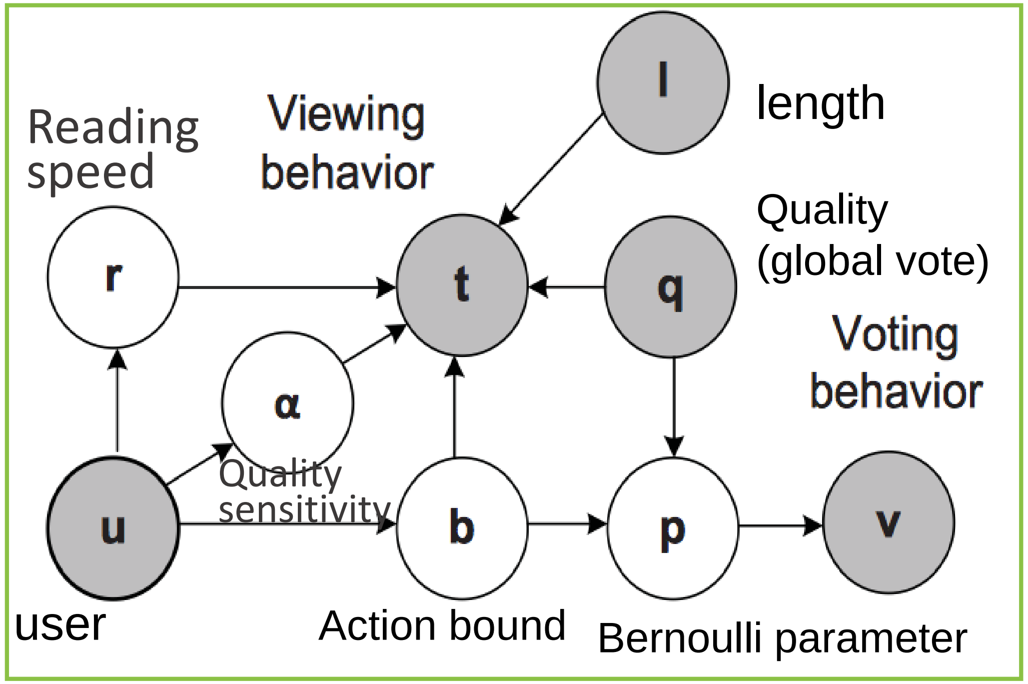


Figure 3: Trend of dwell time w.r.t item length

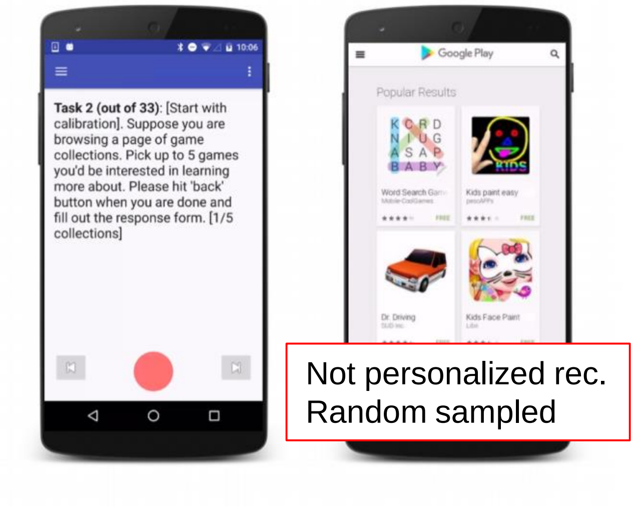
Convert dwell time to "pseudo votes" to enrich the voting matrix for MF



[Yin P, Luo P, et al. Silence is also evidence: interpreting dwell time for recommendation from psychological perspective. In KDD '13]

3. Gaze vs. preference

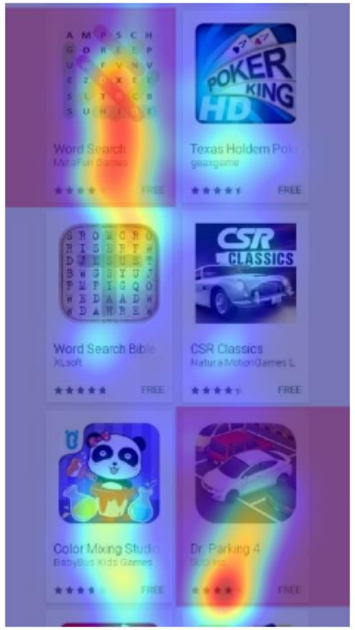
Free browsing; pick up 5 items

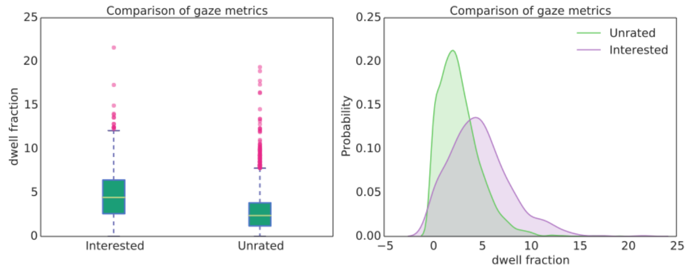


(a) Task description (b) Collection page

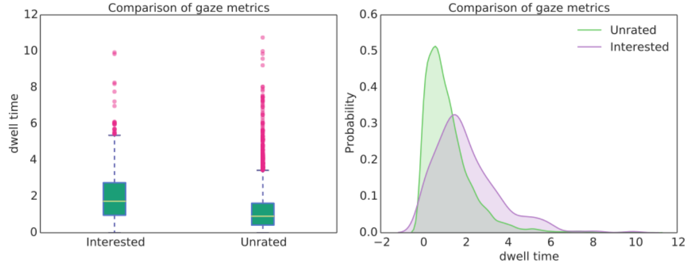
Not personalized rec.
Random sampled

Figure 1: User study app interface. (a): The interface for task instruction. Participants are able to navigate through tasks by pressing "forward" and "backward" button. (b): Example of Google Play Store collection pages.





(a) Gaze dwell fraction

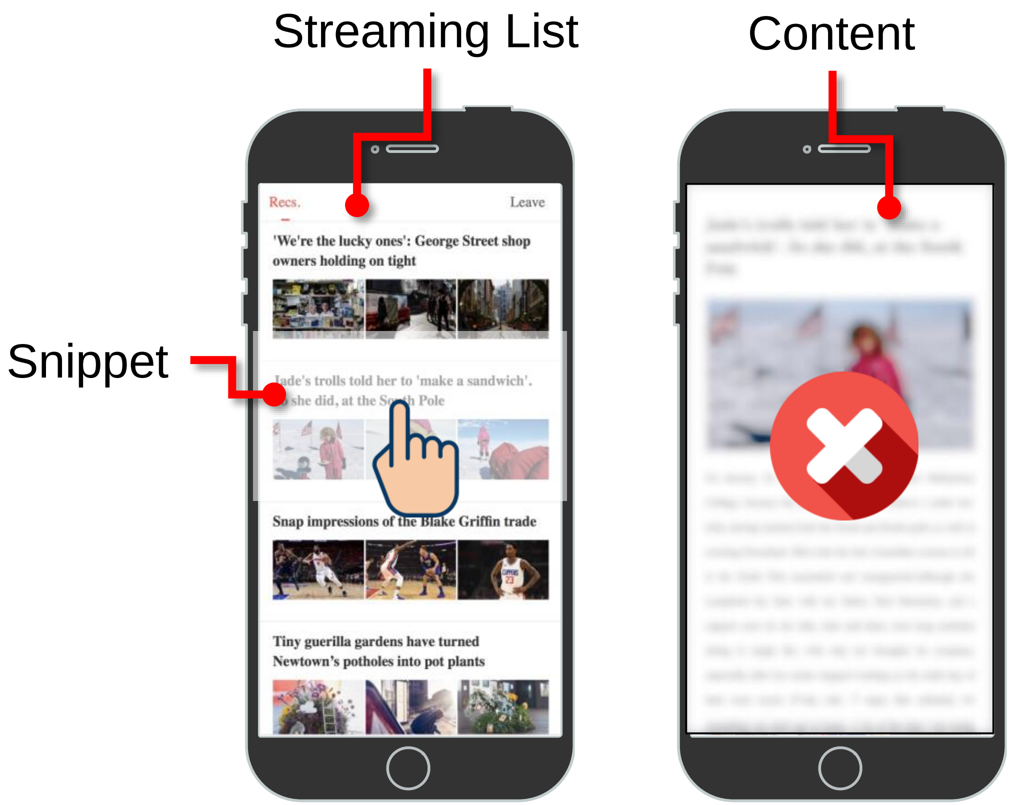


(b) Gaze dwell time

Gaze dwell time is related with preference

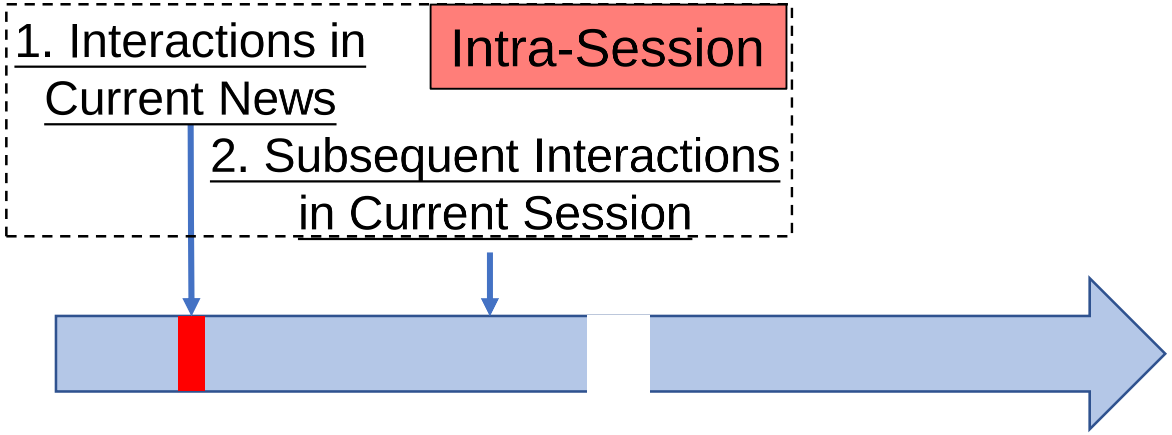
[Li Y, Xu P, et al. Towards measuring and inferring user interest from gaze. In WWW '17]

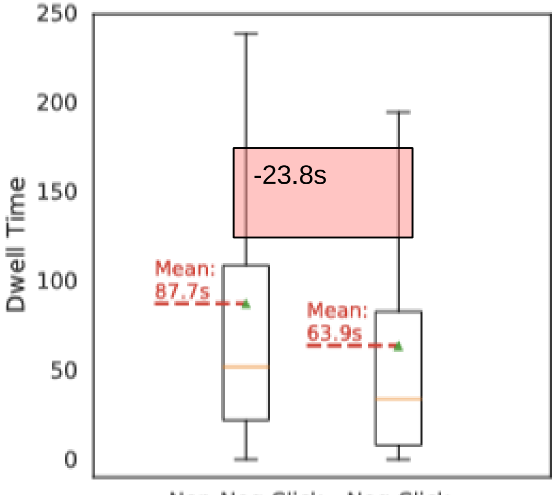
4. Negative Experience



→ Exists negative experience !

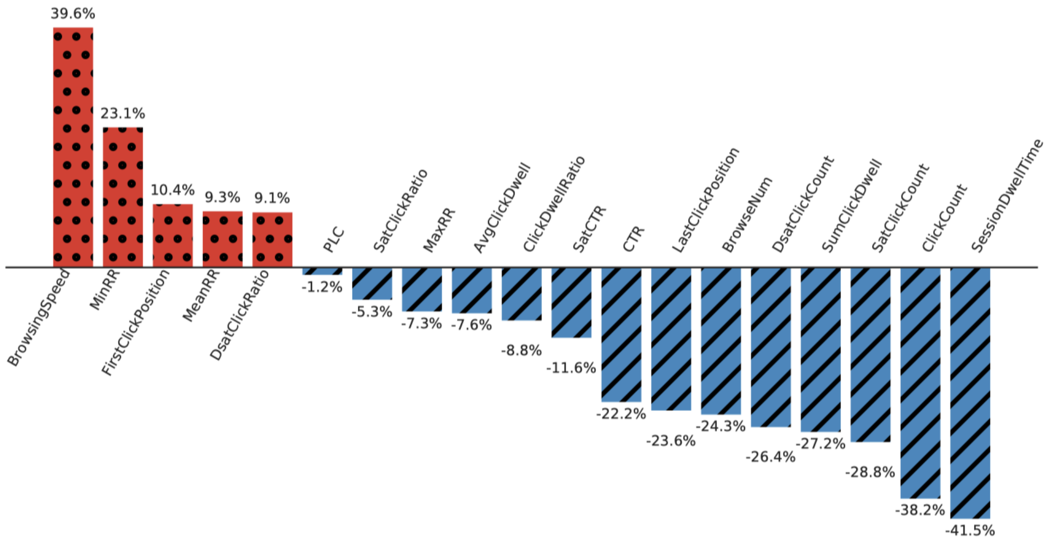
Observations (1):
Intra-Session Effects on User Behaviors





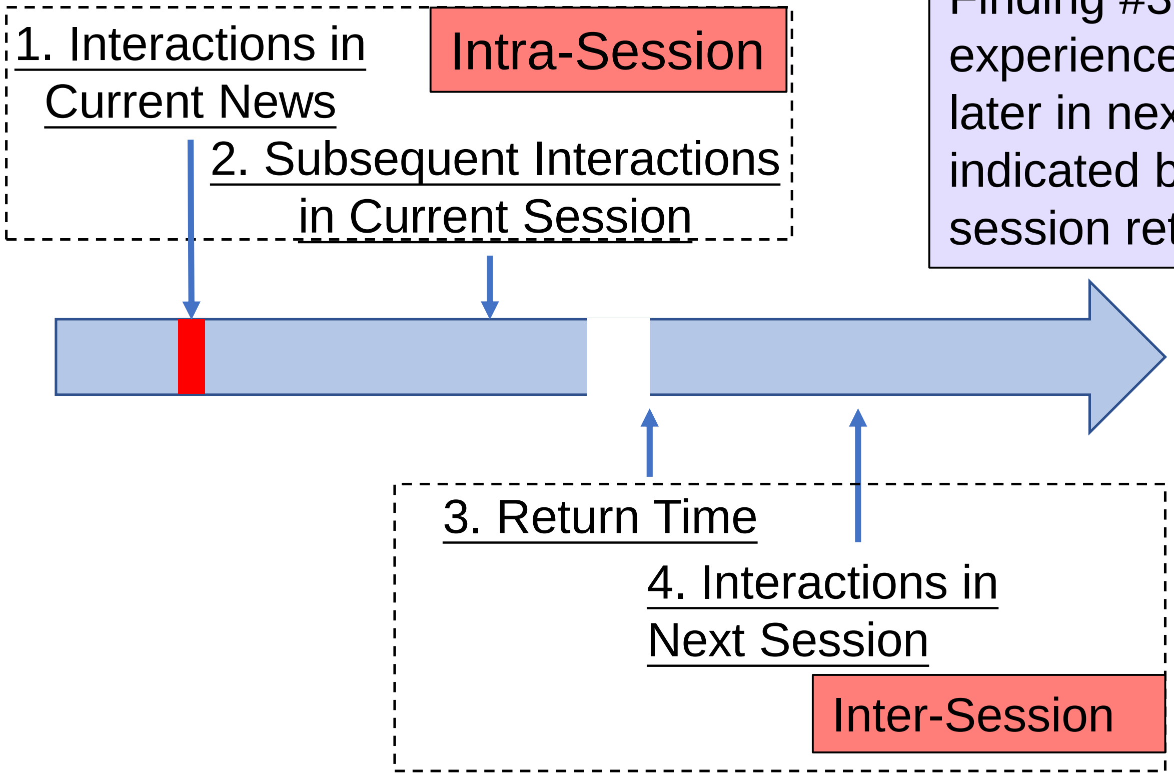
Finding #1: Negative experiences lead to shorter reading, indicated by the shorter dwell time.

Finding #2: After negative experiences, users lose activeness and leave sooner.

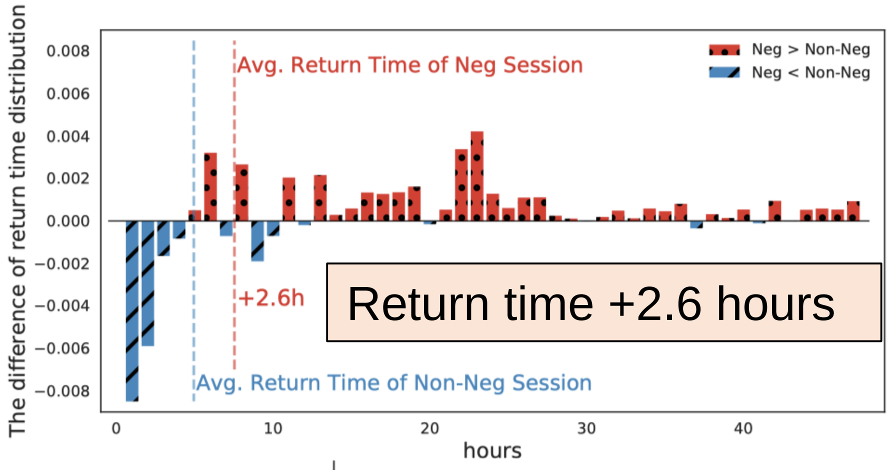


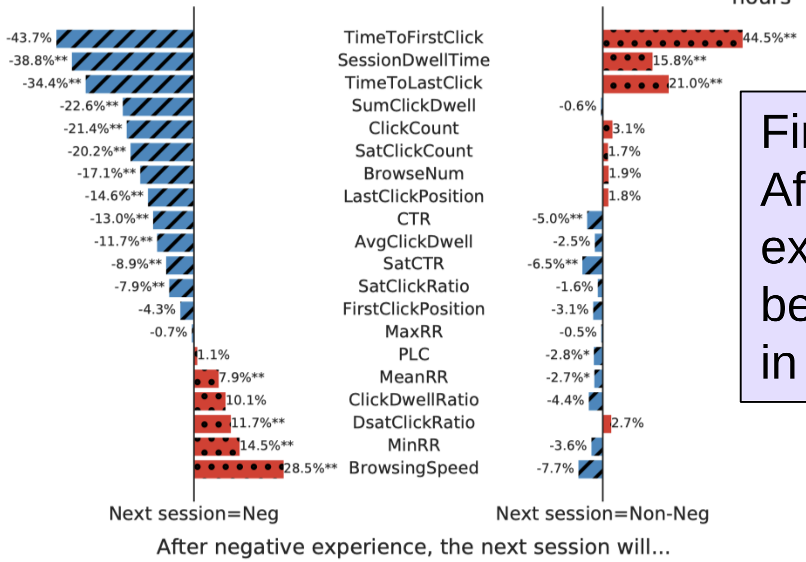
[HY Lu, M Zhang, et al. 2019. Effects of User Negative Experience in Mobile News Streaming. In SIGIR '19]

Observations (2):
Inter-Session Effects on User Behaviors



Finding #3: After negative experiences, user return later in next session, indicated by the longer session return time.





Finding #4:
After negative experiences, users behave differently in the next session.

[HY Lu, M Zhang, et al. 2019. Effects of User Negative Experience in Mobile News Streaming. In SIGIR '19]

Identification & Effects on Satisfaction

Information Retrieval @ Tsinghua University

Negative Experience Identification

Features

F0: Reading interaction
F1: Change of following interactions in Current Session
F2: Return Time
F3: Change of following interactions in Next Session

	Precision	Recall	F-measure	AUC
Sat-Click	0.0147	0.4005	0.0283	0.5433
Lu [8]	0.6740	0.2802	0.3958	0.5723
① F_0	0.6231	0.3615	0.4575	0.5714
② F_0+F_1	0.6379	0.3550	0.4562	0.5767
③ $F_0+F_1+F_2$	0.6499*	0.3941*	0.4906*	0.5909*
④ $F_0+F_1+F_2+F_3$	**0.7729****	**0.4685****	**0.5834****	**0.6654****

Finding #5: Changes of users' subsequent behaviors are useful to identify the negative experiences

[HY Lu, M Zhang, et al. Effects of User Negative Experience in Mobile News Streaming. In SIGIR '19]

Effects on user satisfaction

	User Study	Curr-FirstPos (Fig.4)	Curr-Match (Tab.1)	Next-Neg (Fig.6a)	Next-NonNeg (Fig.6b)
BrowseNum	-	⬇	⬇	⬇	⇧
ClickCount	☻	⬇	⬇	⬇	⇧
CTR	☻	⬇	⬇	⬇	⬇
ClickDwellRatio	☻	⇩			
SessionDwellTime	☻	⬇			
SatClickCount	☻	⬇			
SatCTR	☻	⬇			
SumClickDwell	☻	⬇			
TimeToLastClick	☻	-			
PLC	☻	⇩			
LastClickPosition	☻	⬇			
SatClickRatio	☻	⬇			
MaxRR	☻	⬇			
AvgClickDwell	☻	⇩			
MeanRR	☹	⇧	⇧	⇧	⬇
TimeToFirstClick	☹	-	⇩	⇩	⇧
DsatClickRatio	☹	⇧	⇧	⇧	⇧
FirstClickPosition	☹	⇧	⇩	⬇	⇩
MinRR	☹	⇧	⇧	⇧	⇩
BrowsingSpeed	☹	⇧	⇧	⇧	⇩

After negative experience
=>
SAT-behaviors decrease,
DSAT-behaviors increase.
=>
Satisfaction decrease

Finding #6: Negative experiences reduce user satisfaction in current session

Trends

- Towards challenges
  - Cold-start
  - Efficiency
  - Explainability
  - User satisfaction & behavior
  - **Exploration vs. exploitation (Diversity)**
  - Fairness
- Trending techniques
  - Knowledge-aware
  - Sequential Recommendation
  - Reinforcement Learning
- New scenarios

Towards Challenges – Diversity

- Methods:
  - Aggregating multiple recommenders in diversity
  - Reranking recommendation results from the system
  - Taking user's long-term activities and preferences into consideration
- Many metrics.
  - e.g.: intra-list similarity: $div(R_u) = \sum_{i \in R_u} \sum_{j \in R_u \; i \neq j} d(i,j)$
  - Will see more later

Diversity (1): Aggregating methods

- Re-score function:

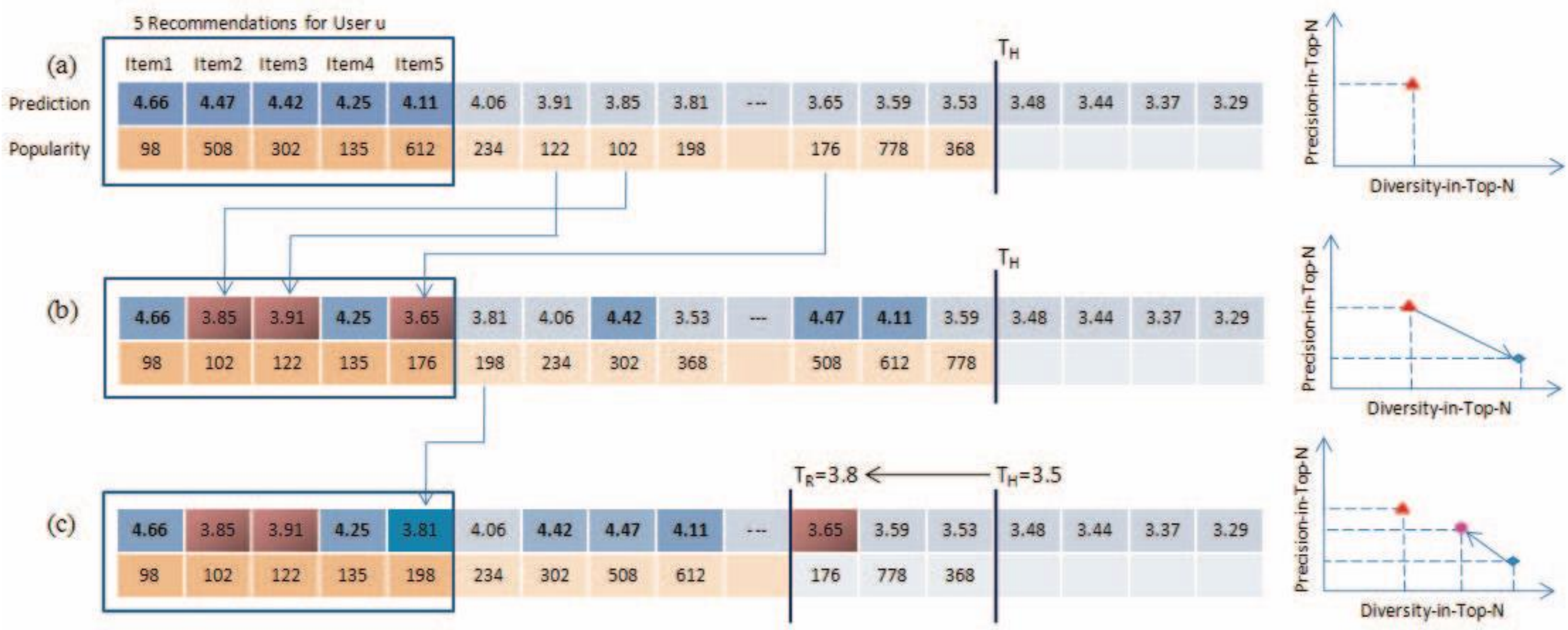
$$rank_x(i, T_R) = \begin{cases} rank_x(i), & \text{if } R^*(u,i) \in [T_R, T_{max}] \\ \alpha_u + rank_{Standard}(i), & \text{if } R^*(u,i) \in [T_H, T_R) \end{cases}$$

$$\text{where } I_u^*(T_R) = \{i \in I \mid R^*(u,i) \geq T_R\}, \alpha_u = \max_{i \in I_u^*(T_R)} rank_x(i).$$

  - T_H: ranking threshold. The rating of each item should not be too low.
  - T_R: re-score threshold. Items with lower scores will be given extra weight.

[Adomavicius G, Kwon Y O. Improving aggregate recommendation diversity using ranking-based techniques. TKDE’2011]

Aggregating Example



Experimental Results

TABLE 3. DIVERSITY GAINS OF PROPOSED RANKING APPROACHES FOR DIFFERENT LEVELS OF PRECISION LOSS

	Item Popularity		Reverse Prediction		Item Average Rating		Item Abs Likeability		Item Relative Likeability		Item Rating Variance		Neighbors' Rating Variance	
Precision Loss	Diversity Gain		Diversity Gain		Diversity Gain		Diversity Gain		Diversity Gain		Diversity Gain		Diversity Gain	
-0.1	+800	3.078	+848	3.203	+975	3.532	+897	3.330	+937	3.434	+386	2.003	+702	2.823
-0.05	+594	2.543	+594	2.543	+728	2.891	+642	2.668	+699	2.816	+283	1.735	+451	2.171
-0.025	+411	2.068	+411	2.068	+513	2.332	+445	2.156	+484	2.257	+205	1.532	+258	1.670
-0.01	+270	1.701	+234	1.608	+311	1.808	+282	1.732	+278	1.722	+126	1.327	+133	1.345
-0.005	+189	1.491	+173	1.449	+223	1.579	+196	1.509	+199	1.517	+91	1.236	+87	1.226
-0.001	+93	1.242	+44	1.114	+78	1.203	+104	1.270	+96	1.249	+21	1.055	+20	1.052
Standard:0.892	385	1.000	385	1.000	385	1.000	385	1.000	385	1.000	385	1.000	385	1.000

(a) MovieLens dataset, top-5 items, heuristic-based technique (item-based CF, 50 neighbors)

	Item Popularity		Reverse Prediction		Item Average Rating		Item Abs Likeability		Item Relative Likeability		Item Rating Variance	
Precision Loss	Diversity Gain		Diversity Gain		Diversity Gain		Diversity Gain		Diversity Gain		Diversity Gain	
-0.1	+314	1.356	+962	2.091	+880	1.998	+732	1.830	+860	1.975	+115	1.130
-0.05	+301	1.341	+757	1.858	+718	1.814	+614	1.696	+695	1.788	+137	1.155
-0.025	+238	1.270	+568	1.644	+535	1.607	+464	1.526	+542	1.615	+110	1.125
-0.01	+156	1.177	+363	1.412	+382	1.433	+300	1.340	+385	1.437	+63	1.071
-0.005	+128	1.145	+264	1.299	+282	1.320	+247	1.280	+288	1.327	+47	1.053
-0.001	+64	1.073	+177	1.201	+118	1.134	+89	1.101	+148	1.168	+8	1.009
Standard:0.834	882	1.000	882	1.000	882	1.000	882	1.000	882	1.000	882	1.000

(b) Netflix dataset, top-5 items, model-based technique (matrix factorization CF, *K*=64)

Diversity (2): Reranking

- Music recommendation reranking:
  - Reranking objective function:

$$Obj(c,R) = Sim(c,P) * (1-\lambda) + \lambda * Div_{overall}(c,R)$$

  - Where:

$$Sim(c,P) = Rank(c,O) \qquad Div_{overall}(c,R) = \sum_{i=1,2,\ldots,n} \theta_i * Div_i(c,R)$$

  - Definition of Div() here:

$$Div_i(c,R) = \frac{\sum_{k \in R} 1 - similarity(c,k)}{n}$$

[Lu F, Tintarev N. A Diversity Adjusting Strategy with Personality for Music Recommendation. IntRS @ Recsys'2018]

Experimental Results

- Evaluation:
  - 25 participants
  - 3 aspects: quality, perceived diversity, and user satisfaction
- Reranked list outperforms the primary list in all aspects.

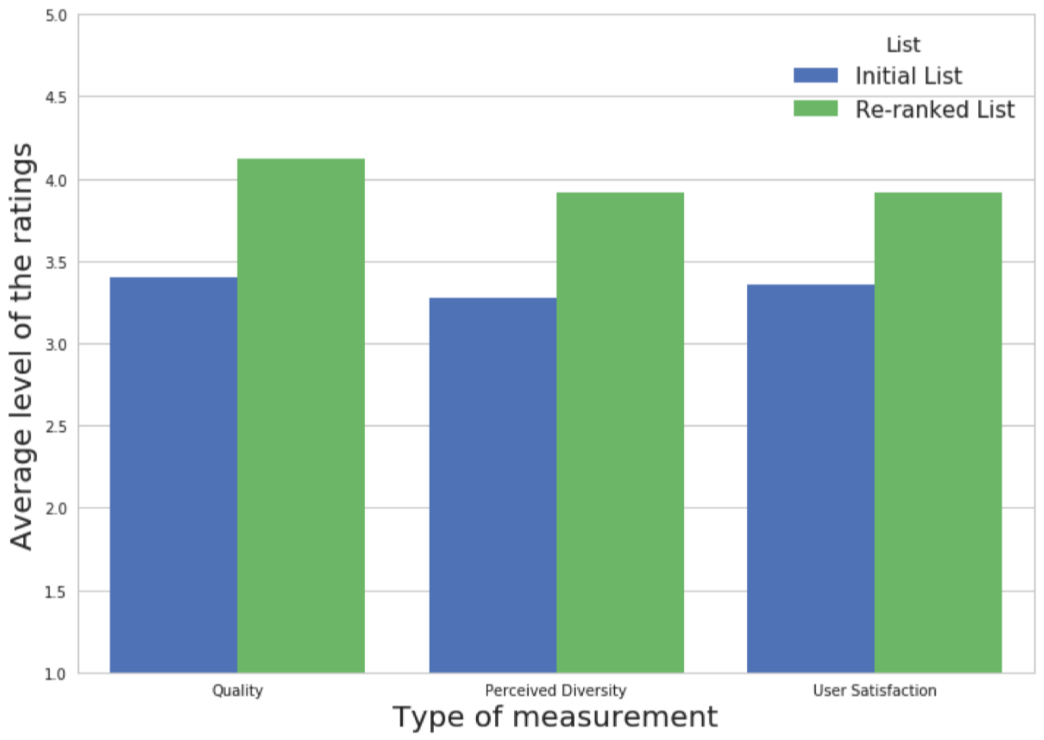
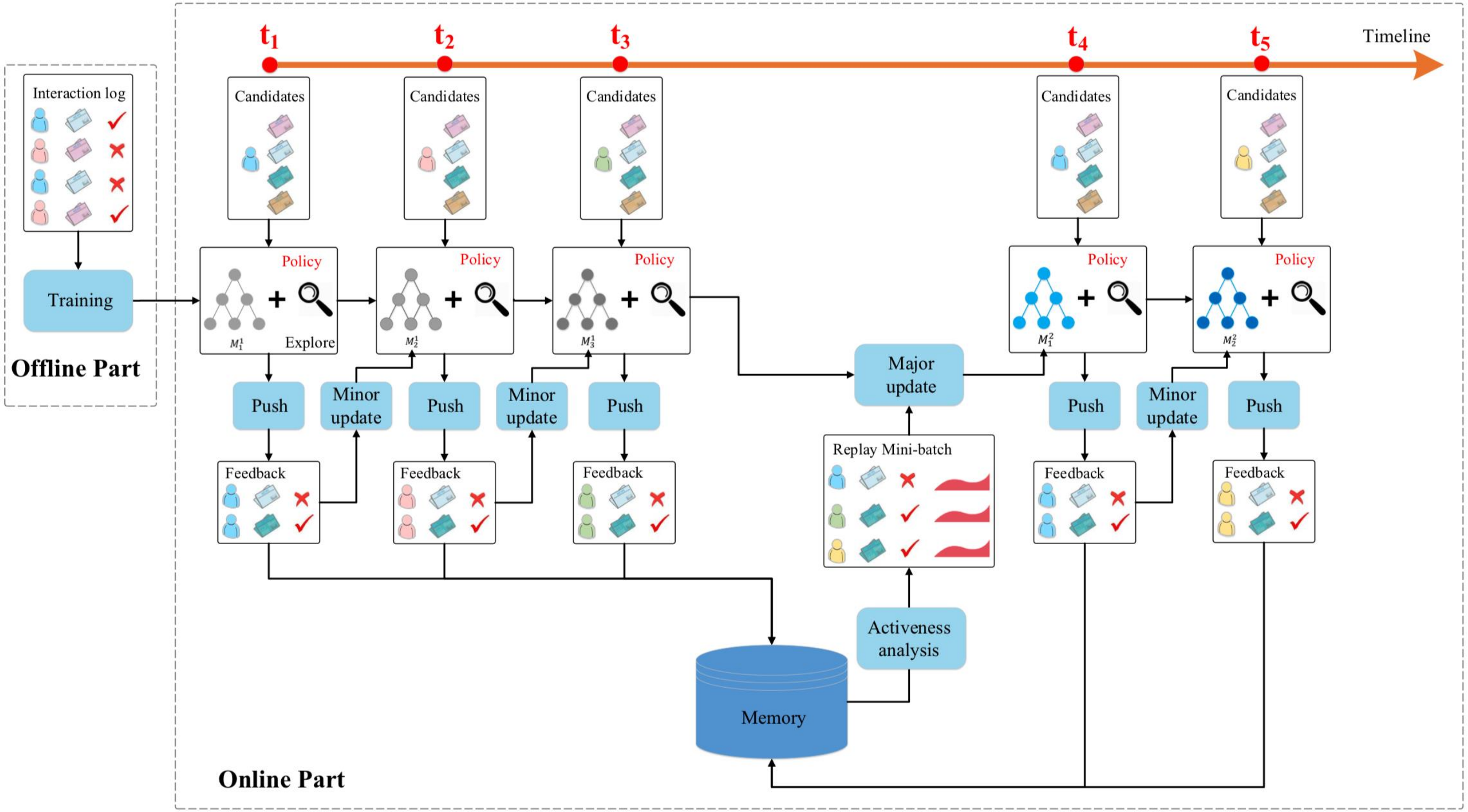


Figure 3: Full comparison for Recommendation Quality (Accuracy), Diversity and User Satisfaction. Student t-Test is also used. $p < 0.05$.

Diversity (3): Using long-term Interests



[Zheng G, Zhang F, Zheng Z, et al. DRN: A deep reinforcement learning framework for news recommendation. WWW’2018]

Experimental Results

Diversity metric: ILS (smaller is better on diversity)

Online recommendation accuracy

Method	CTR	Precision@5	nDCG
LR	0.0059	0.0082	0.0326
FM	0.0072	0.0078	0.0353
W&D	0.0052	0.0067	0.0258
LinUCB	0.0075	0.0091	0.0383
HLinUCB	0.0085	0.0128	0.0449
DN	0.0100	0.0135	0.0474
DDQN	0.0111	0.0139	0.0477
DDQN + U	0.0089	0.0110	0.0425
DDQN + U + EG	0.0083	0.0100	0.03391
DDQN + U + DBGD	**0.0113**	**0.0149**	**0.0492**

$$ILS(\mathrm{L}) = \frac{\sum_{b_i \in \mathrm{L}} \sum_{b_j \in \mathrm{L}, b_j \neq b_i} S(b_i, b_j)}{\sum_{b_i \in \mathrm{L}} \sum_{b_j \in \mathrm{L}, b_j \neq b_i} 1}$$

Method	ILS
LR	0.1833
FM	0.2014
W&D	0.1647
LinUCB	0.2636
HLinUCB	0.1323
DN	0.1546
DDQN	0.1935
DDQN + U	0.1713
DDQN + U + EG	0.1907
DDQN + U + DBGD	**0.1216**

Trends

- Towards challenges
  - Cold-start
  - Efficiency
  - Explainability
  - User satisfaction & behavior
  - Exploration vs. exploitation (Diversity)
  - **Fairness**
- Trending techniques
  - Knowledge-aware
  - Sequential Recommendation
  - Reinforcement Learning
- New scenarios

Towards Challenges – Fairness

"A person's experience with an information system should irrelevantly depend on their personal characteristics (Michael et al. 2018)"

For users

- Model performance is not consistent across users (Michael et al., 2018)
  - Men receive better recommendations
    - MovieLens 1M, LastFM 1K
  - Old (50+) and young (under 18) receive better recommendations
    - LastFM 360K data

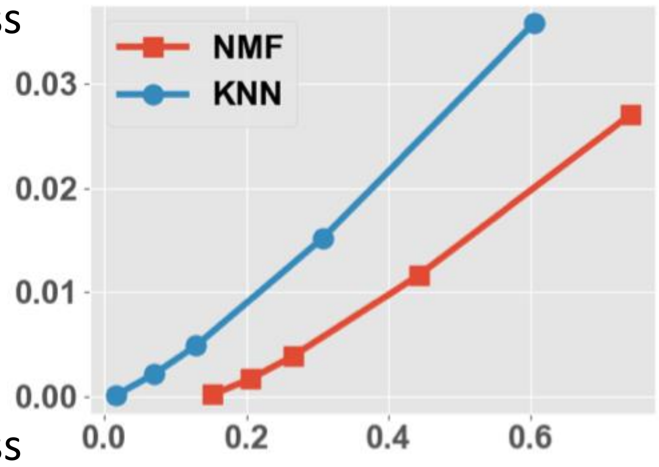
For items

- Skewed distribution of the item exposure (Leonhardt et al., 2018)
  - Most of items are infrequently recommended or not at all

Fairness (1): What causes unfairness

- **Diversity?**
  - Improve diversity
  - → improve the fairness for items
  - → but decrease fairness for users. (Leonhardt et al., 2018)

low fairness

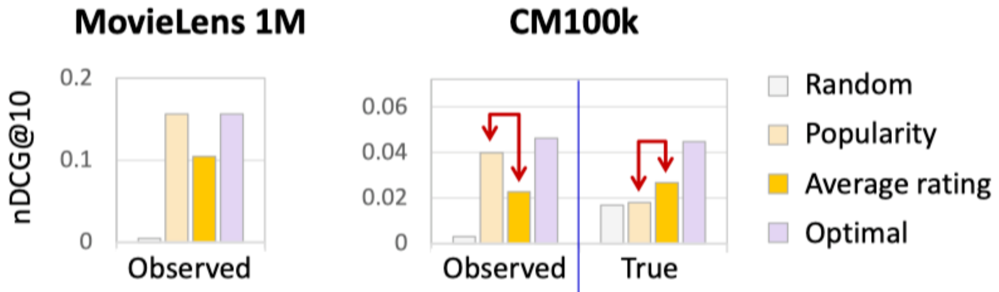


high fairness

low diversity high diversity

- **Popularity?**
  - Traditional offline evaluation methods prefer popular items
  - Lead Recommender systems to only recommend popular items (unfairness)
  - However popularity performs poor in true precision.

(Rocío Cañamares and Pablo Castells, 2018)



- Popularity almost optimal
- Avg rating < popularity

- Popularity ~ random
- Avg rating > popularity

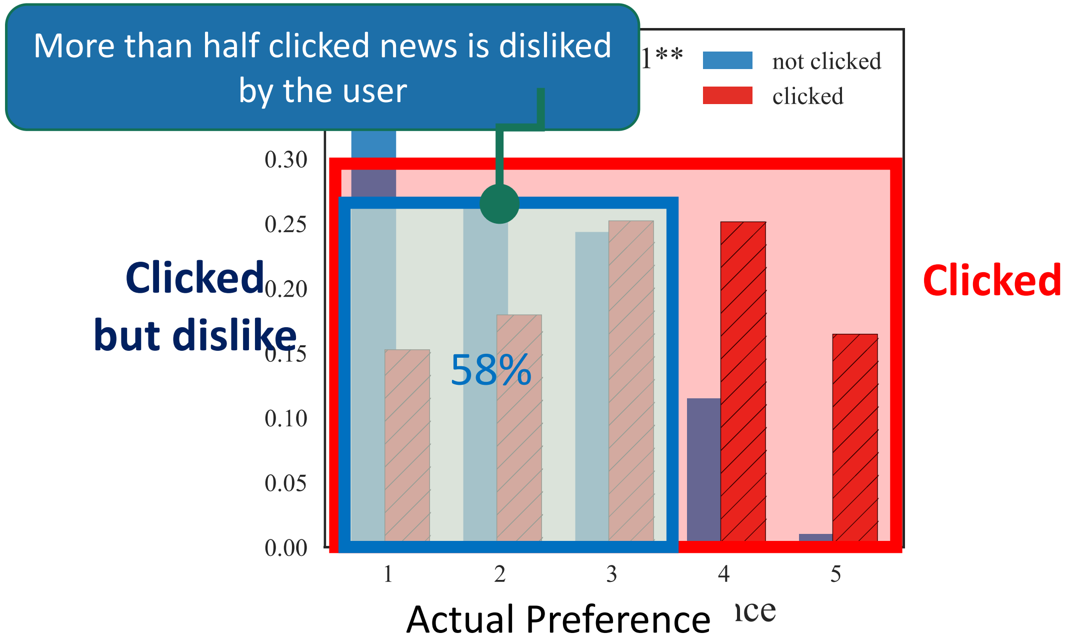
[Leonhardt J, Anand A, Khosla M. User Fairness in Recommender Systems, WWW'18.]

[Cañamares R, Castells P, Should i follow the crowd?: A probabilistic analysis of the effectiveness of popularity in recommender systems, SIGIR'18.]

Fairness (1): What causes unfairness (cont.)

Quality? Low-quality news, like clickbait, receives more clicks

- Click signal ≠ user actual preference. (Lu et al., 2018)
- Reward systems that recommend more low-quality news (unfairness).



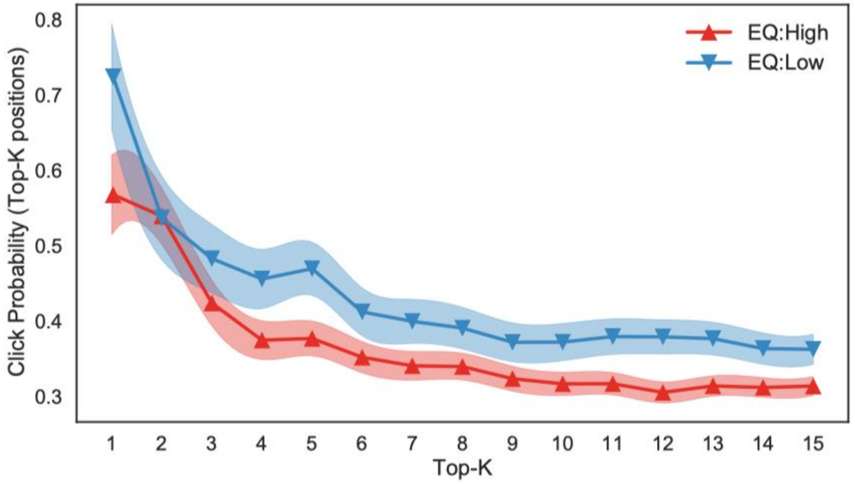


Figure 5: Click Probability of the news in top-k positions conditioned by the news quality. The low-quality news attracts more clicks.

[HY Lu, M Zhang, et al. 2018. Between Clicks and Satisfaction: Study on Multi-Phase User Preferences and Satisfaction for Online News Reading. In SIGIR '18]

Fairness (2): How can we handle it

I. Modeling user preference under unfairness scenarios

- **Behavior to preference ➔ Behavior + quality to preference**
- Improve implicit feedback, by adding two quality effects
  ① Quality affects user preference
  ② Quality affects user behaviors

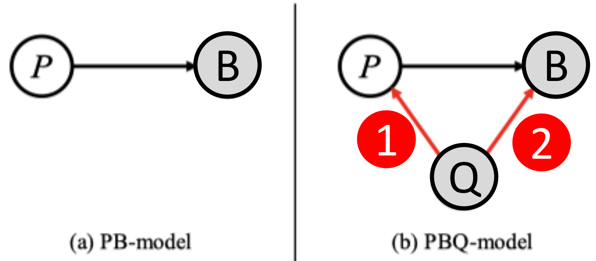


Figure 9: The graph models of traditional implicit feedback which use Behavior to estimate Preference, namely PB-model (a), and implicit feedback which incorporates Quality (b). *P*: preference; *B*: behavior; *Q*: quality.

Table 4: The performance of PB and PBQ with different behavior metrics in estimating user preference.(* represents *p*-value<0.05, ** represents *p*-value<0.01)

Behavior metric	AUC(PB)	AUC(PBQ)	*p*	cohens' *d*
viewport time	0.5775	**0.6249**	**	1.25
dwell time	0.6225[1]	**0.6526**	**	0.88
reading ratio	0.6382	**0.6486**		0.23
reading speed	0.4490	**0.6142**	**	3.32
direction change times	0.5904	**0.6477**	**	1.17
number of interval	0.6111	**0.6709**	**	1.33

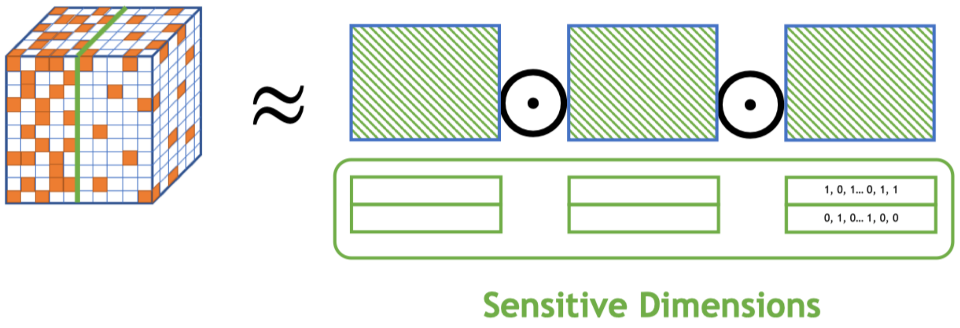
[1] Sat-click, the widely used implicit feedback, can be interpreted as dwell time-based PB-model.

[HY Lu, M Zhang, et al. 2019.Quality Effects on User Preferences and Behaviors in Mobile News Streaming. In WWW '19]

Fairness (2): How can we handle it (cont.)

II. Incorporate fairness into recommendation algorithms

- **Re-ranking**
  - Greedy to choose items to keep the distribution of demographics (Chen Karako and Putra Manggala, 2018)
- **Tensor-based recommend** (Zhu et al. 2018)
  - Isolate sensitive attributes



- **Optimization function** (Lin et al. 2017)
  - Fairness-aware group recommendation

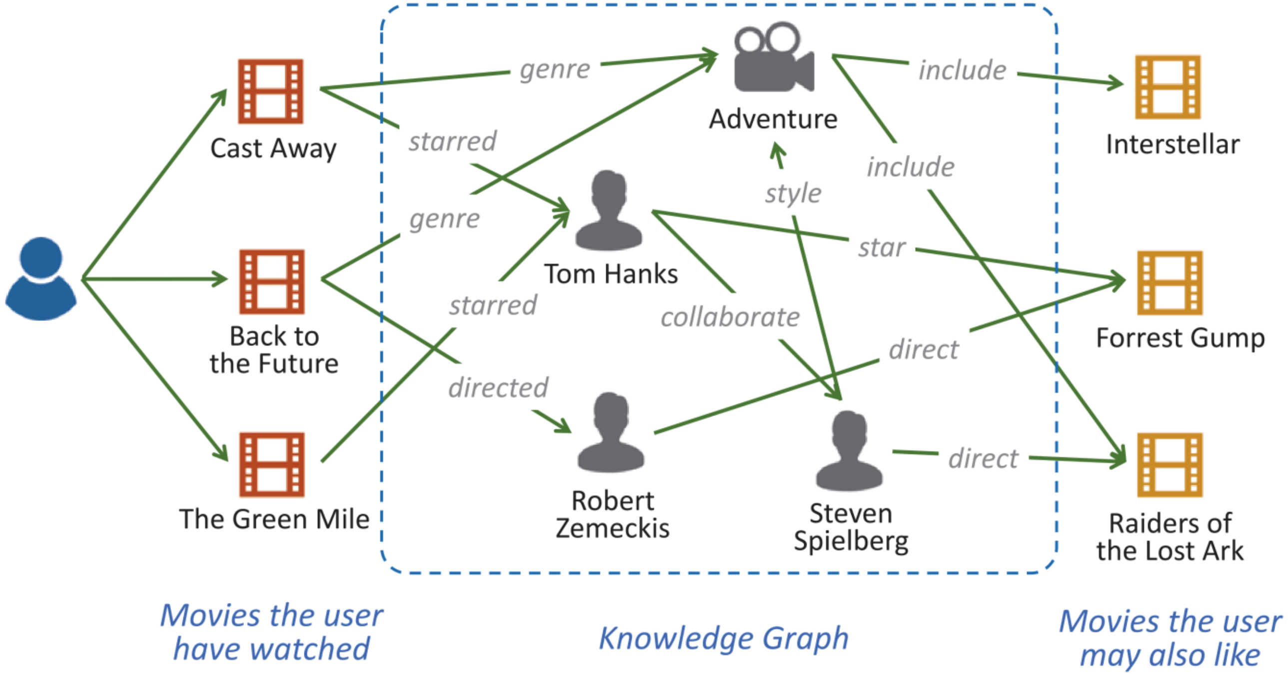
[Lin Xiao, Zhang Min, et al, 2017, Fairness-Aware Group Recommendation with Pareto-Efficiency, RecSys'17.]

Trends

- Towards challenges
  - Cold-start
  - Efficiency
  - Explainability
  - User satisfaction & behavior
  - Exploration vs. exploitation (Diversity)
  - Fairness
- **Trending techniques**
  - **Knowledge-aware**
  - Sequential Recommendation
  - Reinforcement Learning
- New scenarios

Trending techniques (1): Knowledge-aware

- Knowledge Graph: a multi-relational graph that composed of entities as nodes and relations as edges.



[Wang, CIKM 2018]

Trending techniques – Knowledge-aware

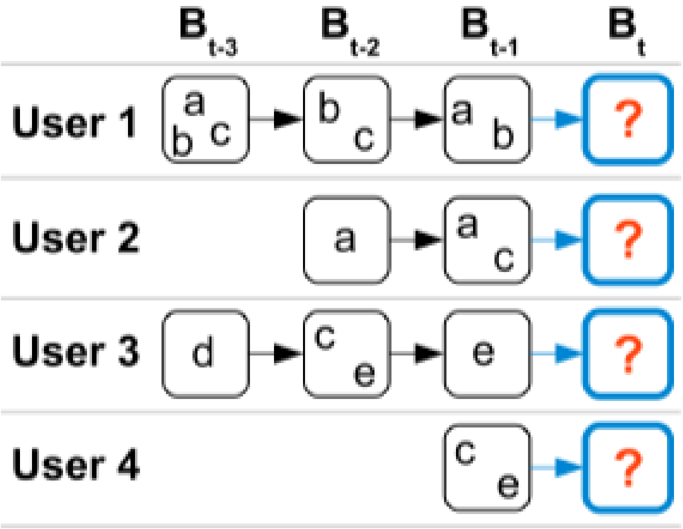
- Various KG resources (general):
  - Freebase (google), DBPedia (from Wikipedia), …
- Contributions:
  - Using KG in embedding (item or user) learning: better performance
  - Using KG in Reasoning (meta-path or rule): better explainability
  - Using KG in Recommendation: helpful for cold items
- Approaches
  - DKN: KG for News Recommendation [Wang H, WWW’18]
  - KPRN: KG for Music Recommendation [Wang X, AAAI’19]
  - RippleNet: Preference Propagation in KG [Wang H, CIKM’18]
  - KGAT: Graph Neural Network with KG [Wang X, KDD’19]
  - RuleRec: KG for Item Recommendation [Ma WZ, WWW’19]
  - ……

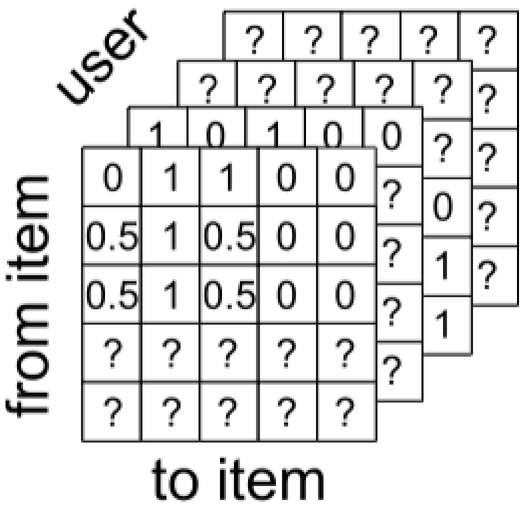
Trends

- Towards challenges
  - Cold-start
  - Efficiency
  - Explainability
  - User satisfaction & behavior
  - Exploration vs. exploitation (Diversity)
  - Fairness
- **Trending techniques**
  - Knowledge-aware
  - **Sequential Recommendation**
  - Reinforcement Learning
- New scenarios

Trending techniques (2): Sequential Recommendation

- **Personalized Sequential Behavior**
  - Modeling ‘third-order’ interactions between a user, his/her previously visited items, and the next item to consume
  - **User is involved** compared with session-based recommendation

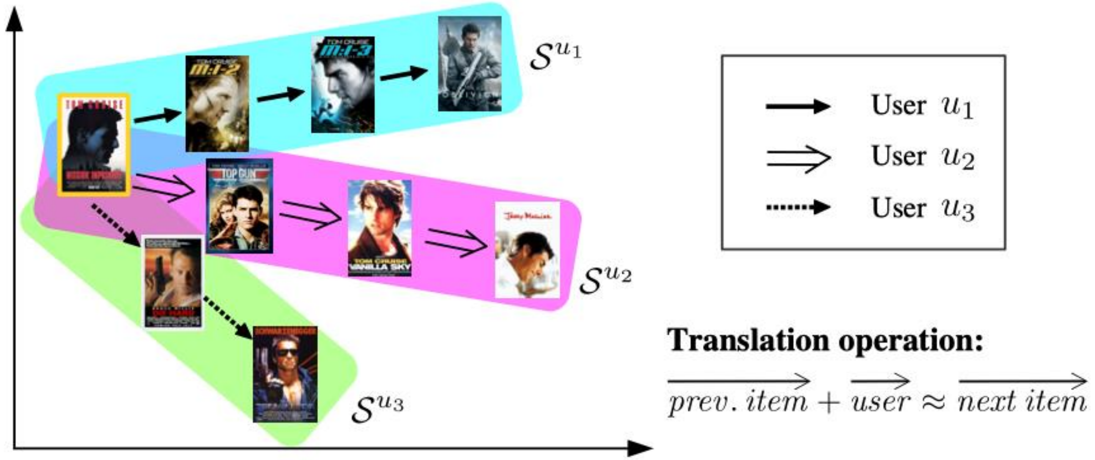




[Rendle S, Freudenthaler C, Schmidt-Thieme L. Factorizing personalized markov chains for next-basket recommendation. WWW 2010.]

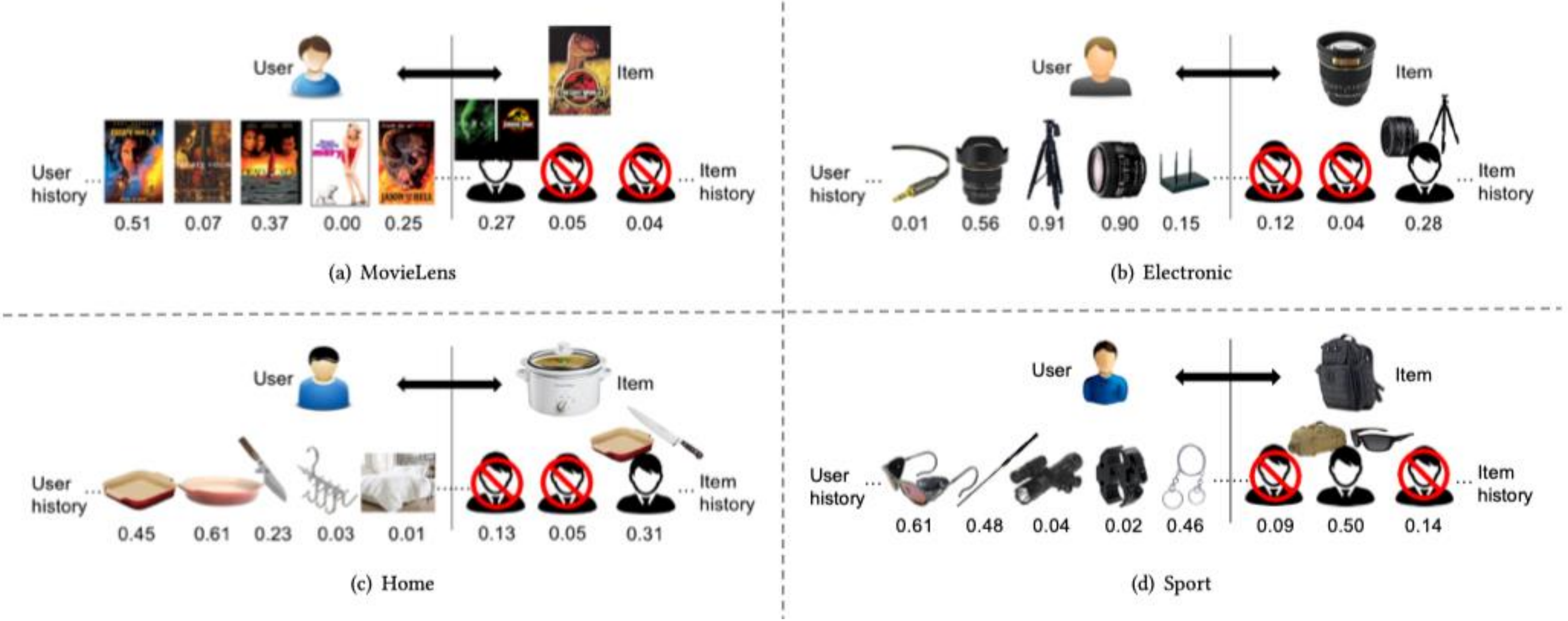
Trending techniques (2): Sequential Recommendation

- **TransRec**: Translation-based Sequential Rec. [He R, RecSys'17]
  - Item – Entity, User – Relation



- **IARN**: Interacting Attention-gated Recurrent Network [Pei W, CIKM'17]
  - Adopt attention mechanism to measure the relevance of each time step
  - Jointly capture user and item dynamics

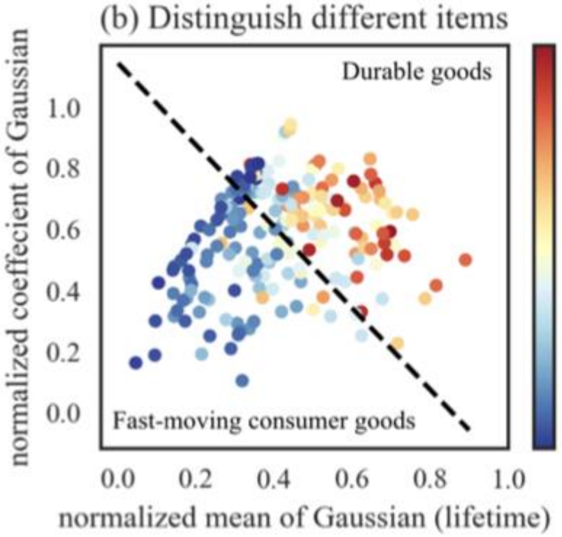
IARN Results Examples

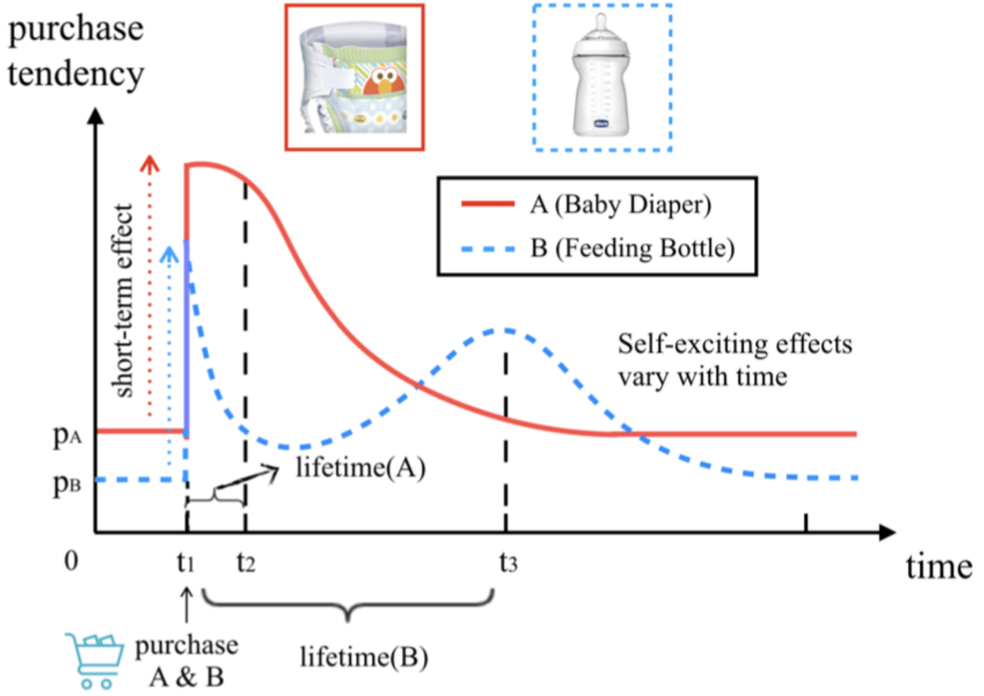


Trending techniques (2): Sequential Recommendation

- SLRC: Take temporal dynamics of *repeat consumption* into consideration
  - Short-term effect & Life-time effect
  - Combine Hawkes Process and CF

$$\lambda^{u,i}(t) = \overbrace{\lambda_0^{u,i}}^{base} + \alpha_i \overbrace{\sum_{(t',i')\in S_t^u} I(i'=i)\gamma_i(t-t')}^{self-excitation}$$





[Wang C, Zhang M, Ma W, et al. Modeling Item-Specific Temporal Dynamics of Repeat Consumption for Recommender Systems. WWW'19.]

Trends

- Towards challenges
  - Cold-start
  - Efficiency
  - Explainability
  - User satisfaction & behavior
  - Exploration vs. exploitation (Diversity)
  - Fairness
- **Trending techniques**
  - Knowledge-aware
  - Sequential Recommendation
  - **Reinforcement Learning**
- New scenarios

Trending techniques (3): RL for RS

- **Conventional Supervised Learning**
  - Consider the recommendation as a static process
  - Maximize the immediate reward of recommended items
- **Reinforcement Learning**
  - Model the interactions as a Markov Decision Process (MDP)
  - Maximize the long term rewards
  - Update strategies during the interactions

Trending techniques (3): RL for RS -- Approaches

- **DEERS** [Zhao X, KDD’18]
  - Capture both positive and negative feedback
  - Add the pairwise regularization term in the loss function
- **FeedRec** [Zou L, KDD’19]
  - Q-Network: leverage hierarchical RNNs to model complex user behaviors
  - S-Network: simulate the environment and assist the Q-Network
- **PGPR**: Policy-Guided Path Reasoning [Xian Y, SIGIR’19]
  - Cast the recommendation problem as a deterministic MDP over the KG

PGPR: Cases

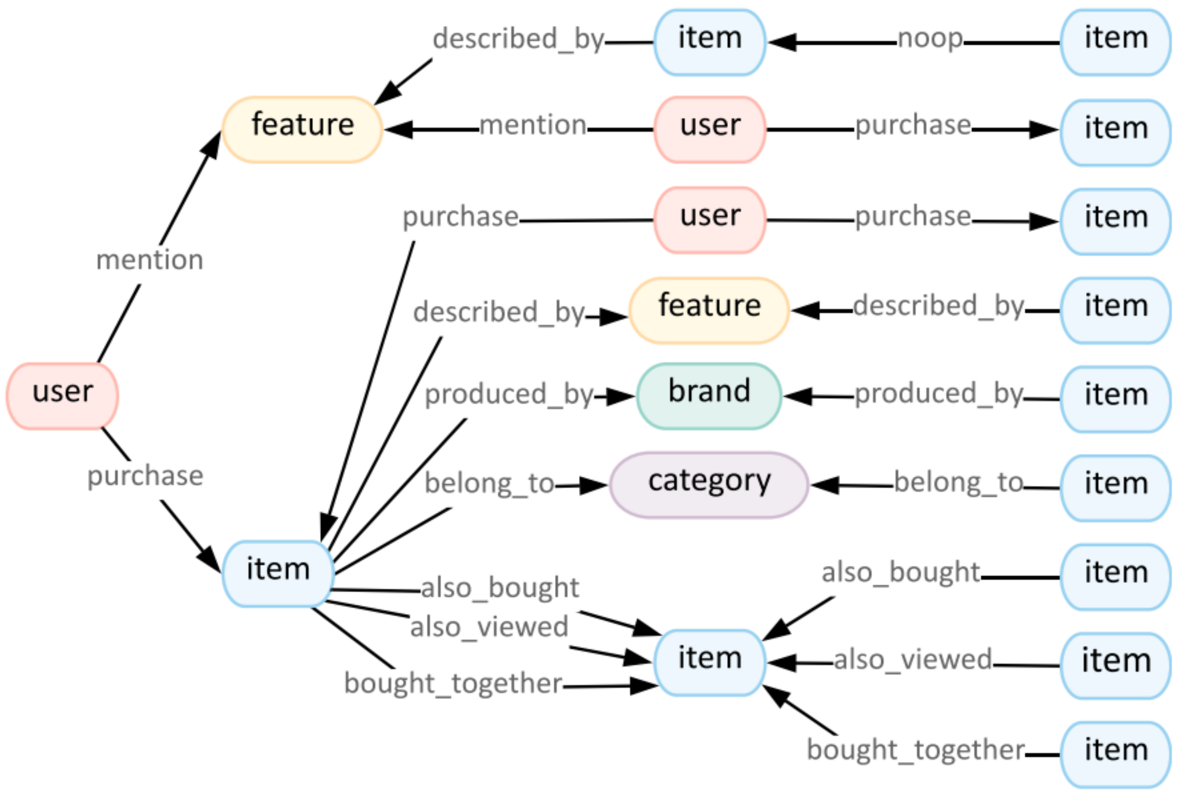


Figure 5: All 3-hop path patterns found in the results.

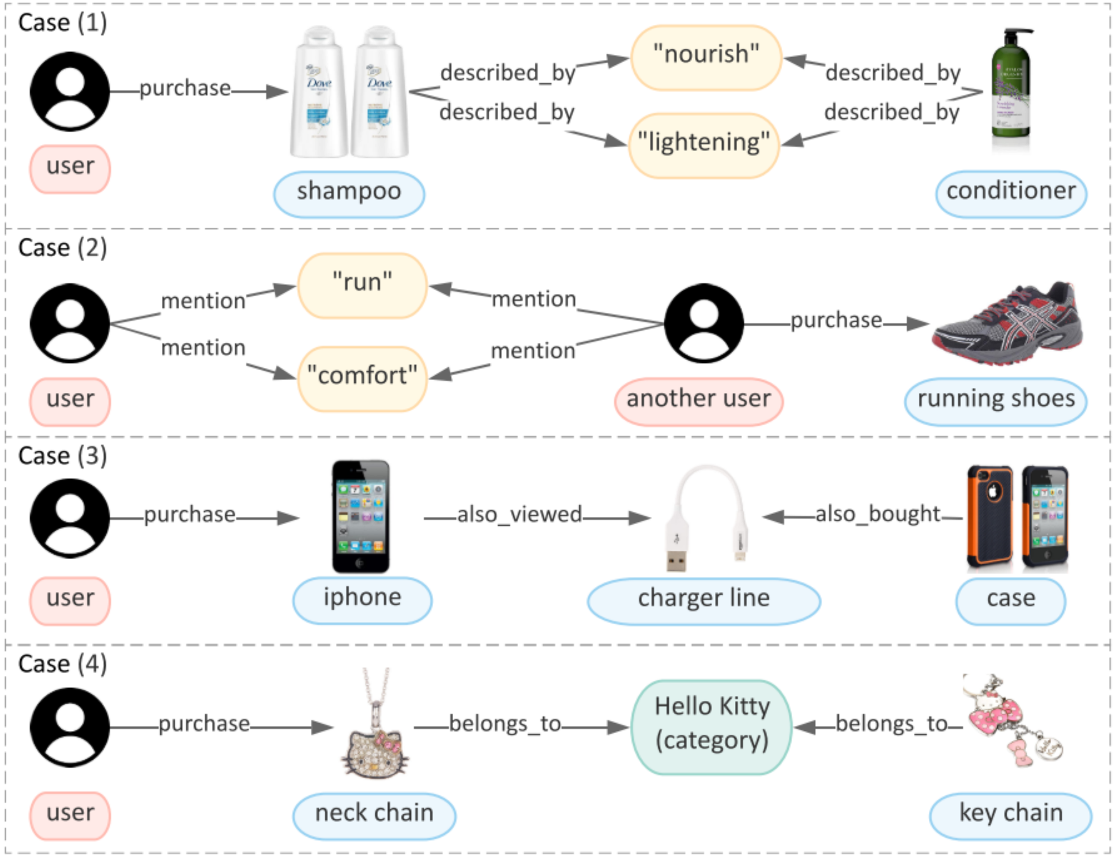


Figure 6: Real cases of recommendation reasoning paths.

Trends

- Towards challenges
  - Cold-start
  - Efficiency
  - Explainability
  - User satisfaction & behavior
  - Exploration vs. exploitation (Diversity)
  - Fairness
- Trending techniques
  - Knowledge-aware
  - Sequential Recommendation
  - Reinforcement Learning
- **New scenarios**

New Scenarios (1) – Law System

- Besides news, products, movies, there are new recommendation scenarios, such as: Law System
- Main differences from news recommendation:
  - User roles should be taken into consideration
    - Judge/Lawyer Vs. Customer
  - Internal relations are more complex
    - Cases/Laws Vs. News

New Scenarios (2) – Knowledge Tracing

- Knowledge tracing: models dynamics in a student's knowledge states in relation to different learning concepts through their interactions with learning activities
- A new task that similar to recommendation
  - User's knowledge state – user preference
  - Interactions with learning activities – interactions in recommender systems
- From recommend items to recommend knowledge

New Scenarios (3): Recommendation in Mobile Environment

- **New information and user interactions:**
  - Different screen size
  - Various time
  - More context features
  - Different user habits/behaviors
  - ……
- **Result Organization:**
  - From cascade to grid layout
  - Less space to show recommendation results

New Scenarios(4): E-health

- Abundant logs provided by different wearable devices (smart watch, smart bracelet, …)
  - Heart rate, sleep quality, steps/day, …
- May corporate with other techniques
  - E.g. Internet of Things (IoT)
- More strict requirements:
  - Higher precision and higher recall

Trends: overview

- Towards challenges
  - Cold-start
  - Efficiency
  - Explainability
  - User satisfaction & behavior
  - Exploration vs. exploitation (Diversity)
  - Fairness
- Trending techniques
  - Knowledge-aware
  - Sequential Recommendation
  - Reinforcement Learning
- **New scenarios**

Summary

Personalized Recommendation:

- Foundations
- Challenges
- Trends

We have passed a long way; but there is still a long way to go.

张敏

z-m@tsinghua.edu.cn

http://www.thuir.cn/group/~mzhang/publications/NLPCC2019-Tutorial.pdf

Reference (1)

- Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems[J]. Computer, 2009 (8): 30-37.
- Rendle S, Freudenthaler C, Gantner Z, et al. BPR: Bayesian personalized ranking from implicit feedback[C]//Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence. AUAI Press, 2009: 452-461.
- Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions[J]. IEEE Transactions on Knowledge & Data Engineering, 2005 (6): 734-749.
- Tan Y, Zhang M, Liu Y, et al. Rating-boosted latent topics: Understanding users and items with ratings and reviews[C]//IJCAI. 2016, 16: 2640-2646.
- Guo C, Zhang M, Liu Y, et al. A picture is worth a thousand words: Introducing visual similarity into recommendation[C]//2016 seventh international conference on intelligent control and information processing (ICICIP). IEEE, 2016: 153-160.

Reference (2)

- Ma W, Zhang M, Wang C, et al. Your Tweets Reveal What You Like: Introducing Cross-media Content Information into Multi-domain Recommendation[C]//IJCAI. 2018: 3484-3490.
- Wang S , Cao L , Wang Y . A Survey on Session-based Recommender Systems[J]. 2019.
- Hidasi B, Karatzoglou A, Baltrunas L, et al. Session-based recommendations with recurrent neural networks[J]. arXiv preprint arXiv:1511.06939, 2015.
- Zhang S, Yao L, Sun A, et al. Deep learning based recommender system: A survey and new perspectives[J]. ACM Computing Surveys (CSUR), 2019, 52(1): 5.
- Cheng H T, Koc L, Harmsen J, et al. Wide & deep learning for recommender systems[C]//Proceedings of the 1st workshop on deep learning for recommender systems. ACM, 2016: 7-10.
- Guo H, Tang R, Ye Y, et al. DeepFM: a factorization-machine based neural network for CTR prediction[J]. arXiv preprint arXiv:1703.04247, 2017.

Reference (3)

- He X, Liao L, Zhang H, et al. Neural collaborative filtering[C]//Proceedings of the 26th international conference on world wide web. International World Wide Web Conferences Steering Committee, 2017: 173-182.
- Huang P S, He X, Gao J, et al. Learning deep structured semantic models for web search using clickthrough data[C]//Proceedings of the 22nd ACM international conference on Information & Knowledge Management. ACM, 2013: 2333-2338.
- Sedhain S, Menon A K, Sanner S, et al. Autorec: Autoencoders meet collaborative filtering[C]//Proceedings of the 24th International Conference on World Wide Web. ACM, 2015: 111-112.
- Shi S, Zhang M, Liu Y, et al. Attention-based adaptive model to unify warm and cold starts recommendation[C]//Proceedings of the 27th ACM International Conference on Information and Knowledge Management. ACM, 2018: 127-136.
- Wang X, He X, Nie L, et al. Item silk road: Recommending items from information domains to social users[C]//Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval. ACM, 2017: 185-194.

Reference (4)

- Song W, Xiao Z, Wang Y, et al. Session-based social recommendation via dynamic graph attention networks[C]//Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. ACM, 2019: 555-563.
- He X, Zhang H, Kan M Y, et al. Fast matrix factorization for online recommendation with implicit feedback[C]//Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. ACM, 2016: 549-558.
- Chen C, Zhang M, Wang C, et al. An Efficient Adaptive Transfer Neural Network for Social-aware Recommendation[C]//Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2019: 225-234.
- Zhang Y, Lai G, Zhang M, et al. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis[C]//Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval. ACM, 2014: 83-92.
- Chen C, Zhang M, Liu Y, et al. Neural attentional rating regression with review-level explanations[C]//Proceedings of the 2018 World Wide Web Conference. International World Wide Web Conferences Steering Committee, 2018: 1583-1592.

Reference (5)

- Chen C, Zhang M, Liu Y, et al. Social attentional memory network: Modeling aspect-and friend-level differences in recommendation[C]//Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. ACM, 2019: 177-185.
- Lu H, Zhang M, Ma S. Between Clicks and Satisfaction: Study on Multi-Phase User Preferences and Satisfaction for Online News Reading[C]//The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. ACM, 2018: 435-444.
- Yin P, Luo P, Lee W C, et al. Silence is also evidence: interpreting dwell time for recommendation from psychological perspective[C]//Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013: 989-997.
- Li Y, Xu P, Lagun D, et al. Towards measuring and inferring user interest from gaze[C]//Proceedings of the 26th International Conference on World Wide Web Companion. International World Wide Web Conferences Steering Committee, 2017: 525-533.
- Lu H, Zhang M, Ma W, et al. Effects of User Negative Experience in Mobile News Streaming[J]. 2019.

Reference (6)

- Zheng G, Zhang F, Zheng Z, et al. DRN: A deep reinforcement learning framework for news recommendation[C]//Proceedings of the 2018 World Wide Web Conference. International World Wide Web Conferences Steering Committee, 2018: 167-176.
- Lu F, Tintarev N. A Diversity Adjusting Strategy with Personality for Music Recommendation[C]//IntRS@ RecSys. 2018: 7-14.
- Adomavicius G, Kwon Y O. Improving aggregate recommendation diversity using ranking-based techniques[J]. IEEE Transactions on Knowledge and Data Engineering, 2011, 24(5): 896-911.
- Leonhardt J , Anand A , Khosla M . User Fairness in Recommender Systems[J]. 2018.
- Cañamares R, Castells P. Should i follow the crowd?: A probabilistic analysis of the effectiveness of popularity in recommender systems[C]//The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. ACM, 2018: 415-424.

Reference (7)

- Lu H, Zhang M, Ma S. Between Clicks and Satisfaction: Study on Multi-Phase User Preferences and Satisfaction for Online News Reading[C]//The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. ACM, 2018: 435-444.
- Lu H, Zhang M, Ma W, et al. Quality Effects on User Preferences and Behaviorsin Mobile News Streaming[C]//The World Wide Web Conference. ACM, 2019: 1187-1197.
- Xiao L, Min Z, Yongfeng Z, et al. Fairness-aware group recommendation with pareto-efficiency[C]//Proceedings of the Eleventh ACM Conference on Recommender Systems. ACM, 2017: 107-115.
- Wang H, Zhang F, Wang J, et al. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems[C]//Proceedings of the 27th ACM International Conference on Information and Knowledge Management. ACM, 2018: 417-426.
- Wang H, Zhang F, Xie X, et al. DKN: Deep knowledge-aware network for news recommendation[C]//Proceedings of the 2018 World Wide Web Conference. International World Wide Web Conferences Steering Committee, 2018: 1835-1844.

Reference (8)

- Ma W, Zhang M, Cao Y, et al. Jointly Learning Explainable Rules for Recommendation with Knowledge Graph[C]//The World Wide Web Conference. ACM, 2019: 1210-1221.
- Wang X, He X, Cao Y, et al. KGAT: Knowledge Graph Attention Network for Recommendation[J]. arXiv preprint arXiv:1905.07854, 2019.
- Rendle S, Freudenthaler C, Schmidt-Thieme L. Factorizing personalized markov chains for next-basket recommendation[C]//Proceedings of the 19th international conference on World wide web. ACM, 2010: 811-820.
- He R, Kang W C, McAuley J. Translation-based recommendation[C]//Proceedings of the Eleventh ACM Conference on Recommender Systems. ACM, 2017: 161-169.
- Pei W, Yang J, Sun Z, et al. Interacting attention-gated recurrent networks for recommendation[C]//Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. ACM, 2017: 1459-1468.

Reference (9)

- Wang C, Zhang M, Ma W, et al. Modeling Item-Specific Temporal Dynamics of Repeat Consumption for Recommender Systems[C]//The World Wide Web Conference. ACM, 2019: 1977-1987.
- Zhao X, Zhang L, Ding Z, et al. Recommendations with negative feedback via pairwise deep reinforcement learning[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2018: 1040-1048.
- Zou L, Xia L, Ding Z, et al. Reinforcement Learning to Optimize Long-term User Engagement in Recommender Systems[J]. arXiv preprint arXiv:1902.05570, 2019.
- Xian Y, Fu Z, Muthukrishnan S, et al. Reinforcement Knowledge Graph Reasoning for Explainable Recommendation[J]. arXiv preprint arXiv:1906.05237, 2019.
- Wang X, Wang D, Xu C, et al. Explainable reasoning over knowledge graphs for recommendation[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2019, 33: 5329-5336.