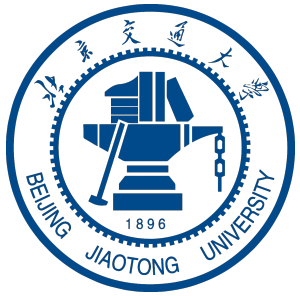
Trustworthy User Modeling

Explainability, Robustness, Fairness, Privacy & Security

Xiaowen Huang

Beijing Jiaotong University

xwhuang@bjtu.edu.cn



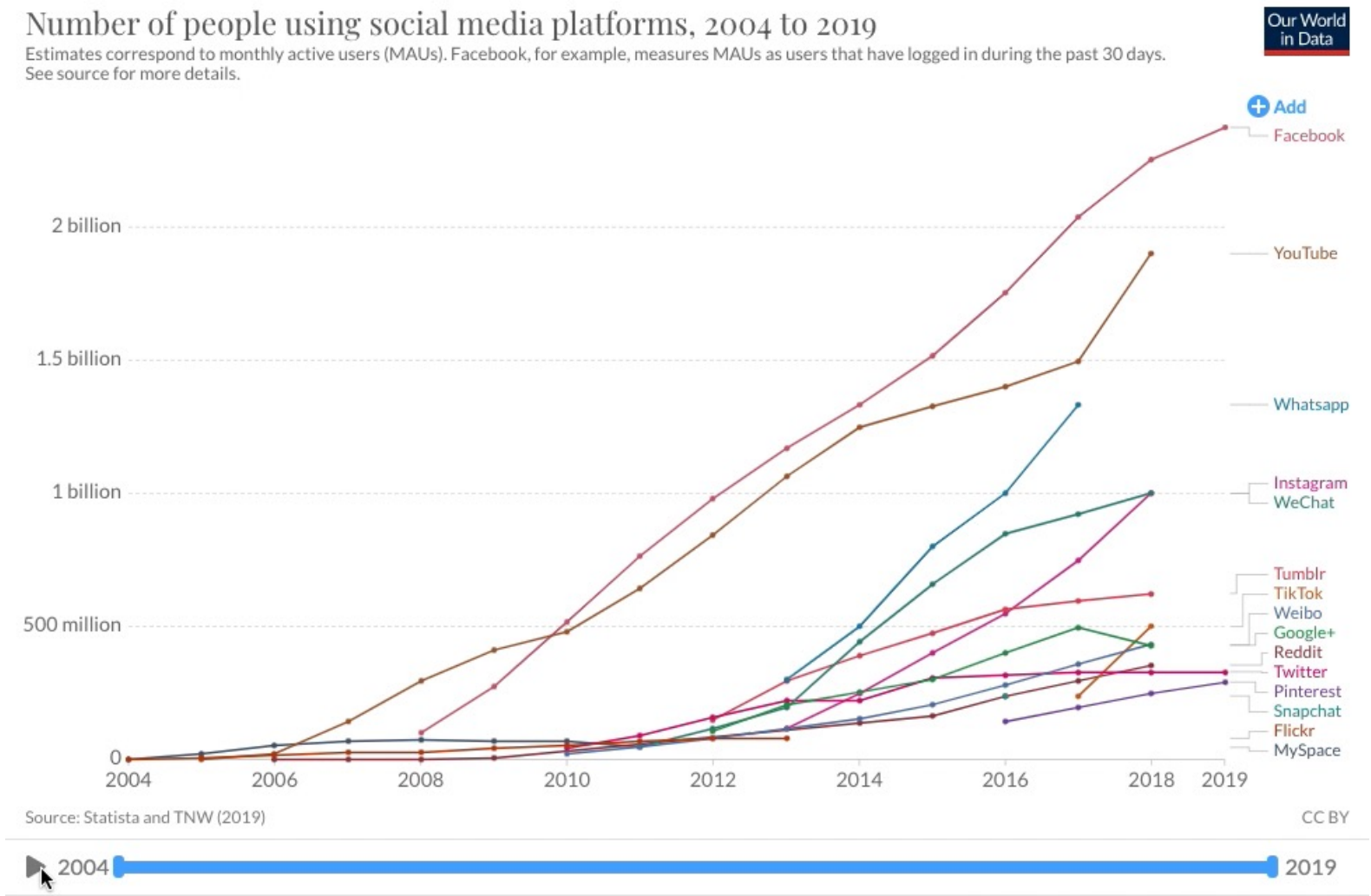
Tutorial website：**https://adam-bjtu.org/**

Social Media

https://ourworldindata.org/rise-of-social-media

The world's most popular social media-MAU （Million） （As of 1/25/2020）	
Facebook	2449
YouTube	2000
Whatsapp	1600
FB message	1300
Wechat	1151
Instagram	1000
Tiktok	800
QQ	731
Qzone	517
Weibo	497
Reddit	430
Snapchat	382
Twitter	340
Pinterest	322
Kuaishou	316

Social Media



Social media started in the early 2000s.

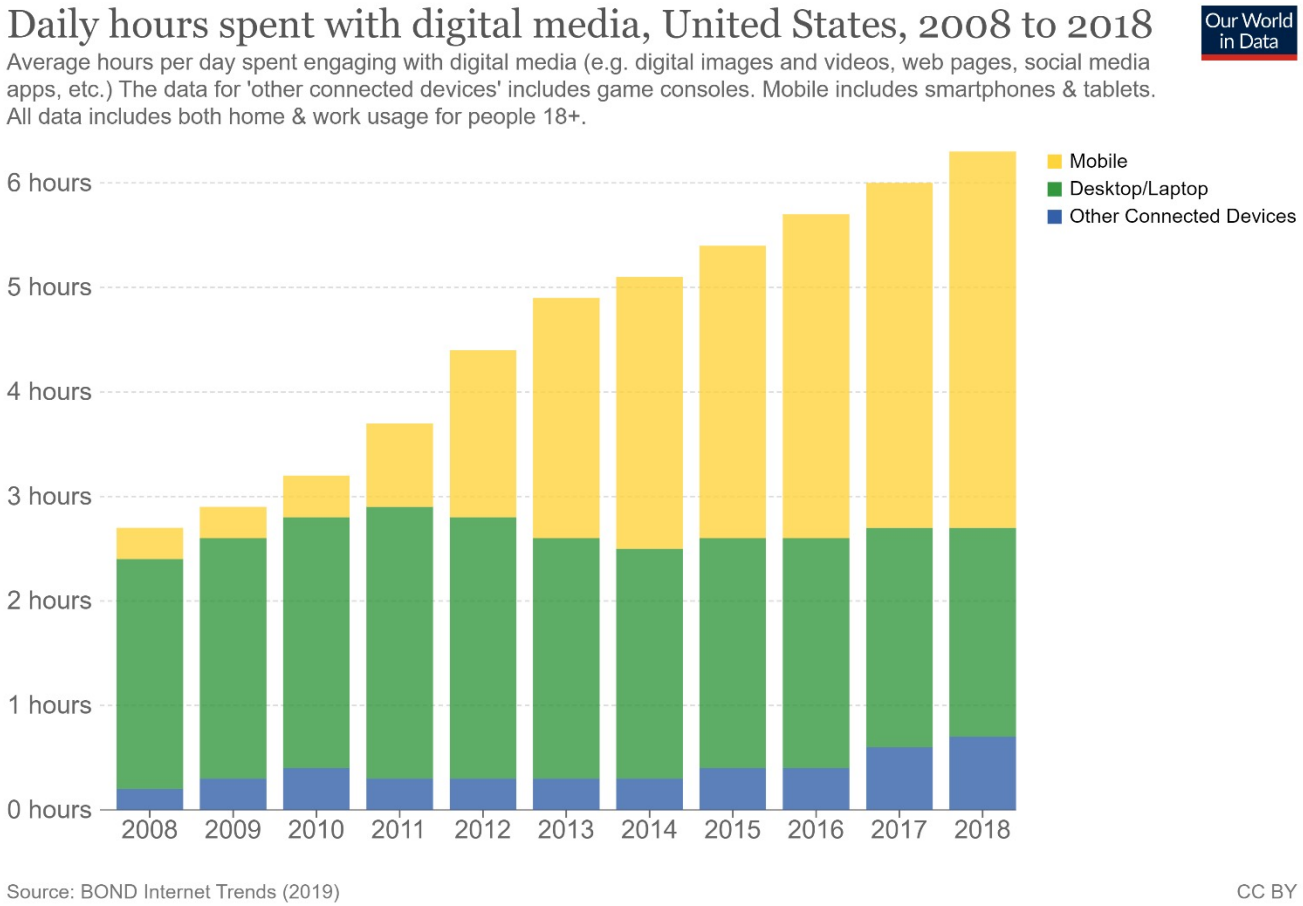
The rise of various social media has also brought about an explosive growth in the number of social media users.

This chart shows that there are some large social media sites that have been around for ten or more years, such as Facebook, YouTube, and Reddit; but other large sites are much newer.

TikTok, for example, launched in September 2016 and by mid-2018 it had already reached half a billion users. To put this in perspective: TikTok gained on average about 20 million new users per month over this period.

https://ourworldindata.org/rise-of-social-media

Social Media

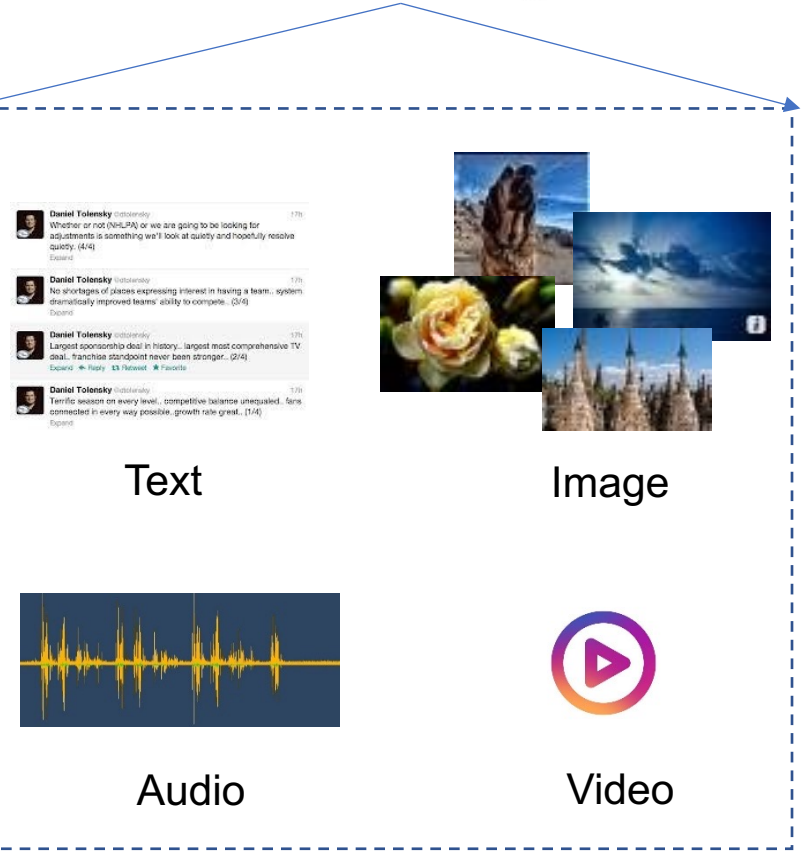


The rise of social media in rich countries has come together with an increase in the amount of time spent online.

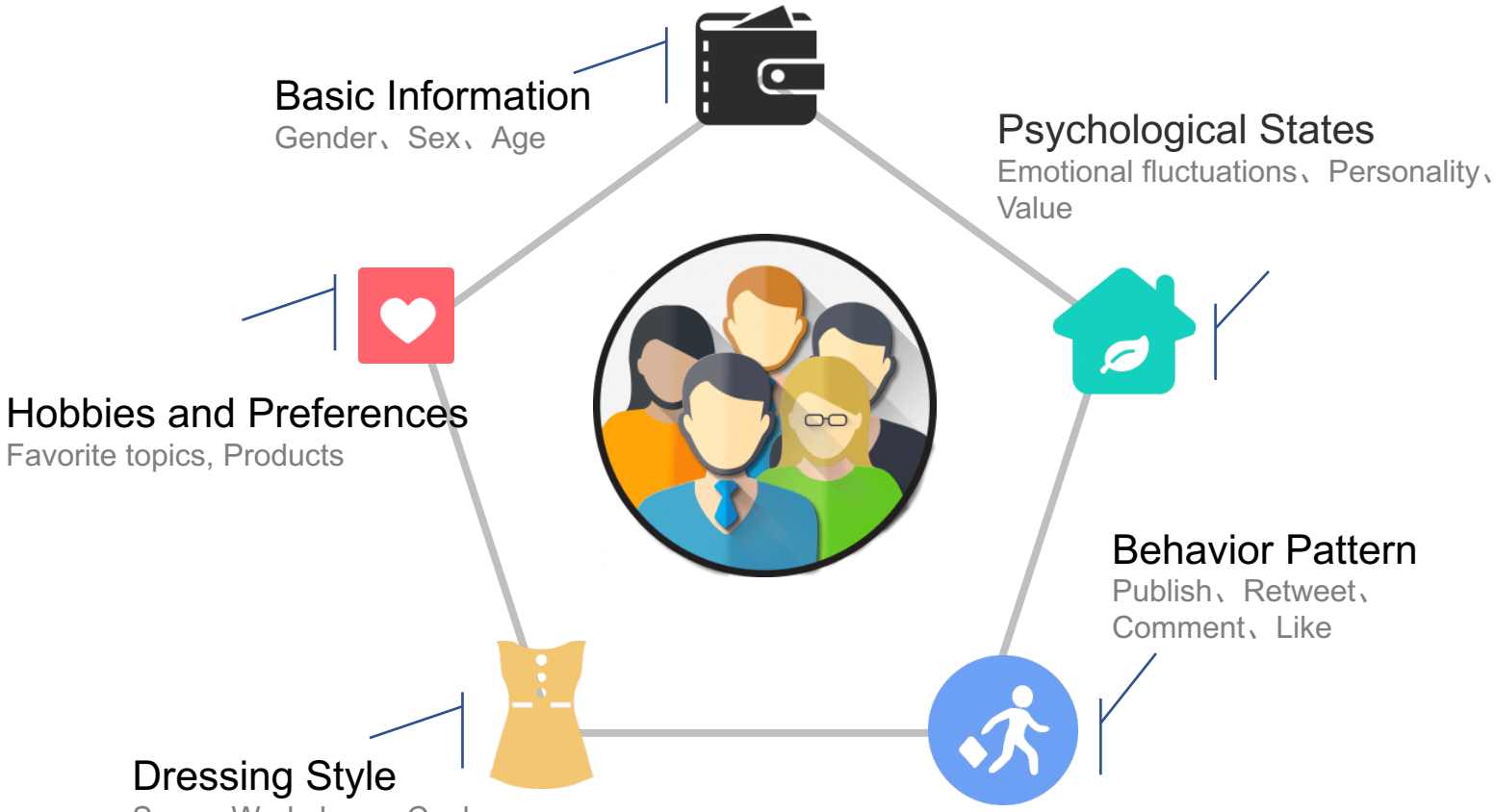
In the US, adults spend more than 6 hours per day on digital media (apps and websites accessed through mobile phones, tablets, computers, and other connected devices such as game consoles). As the chart shows, this growth has been driven almost entirely by additional time spent on smartphones and tablets.

https://ourworldindata.org/rise-of-social-media

User Modeling



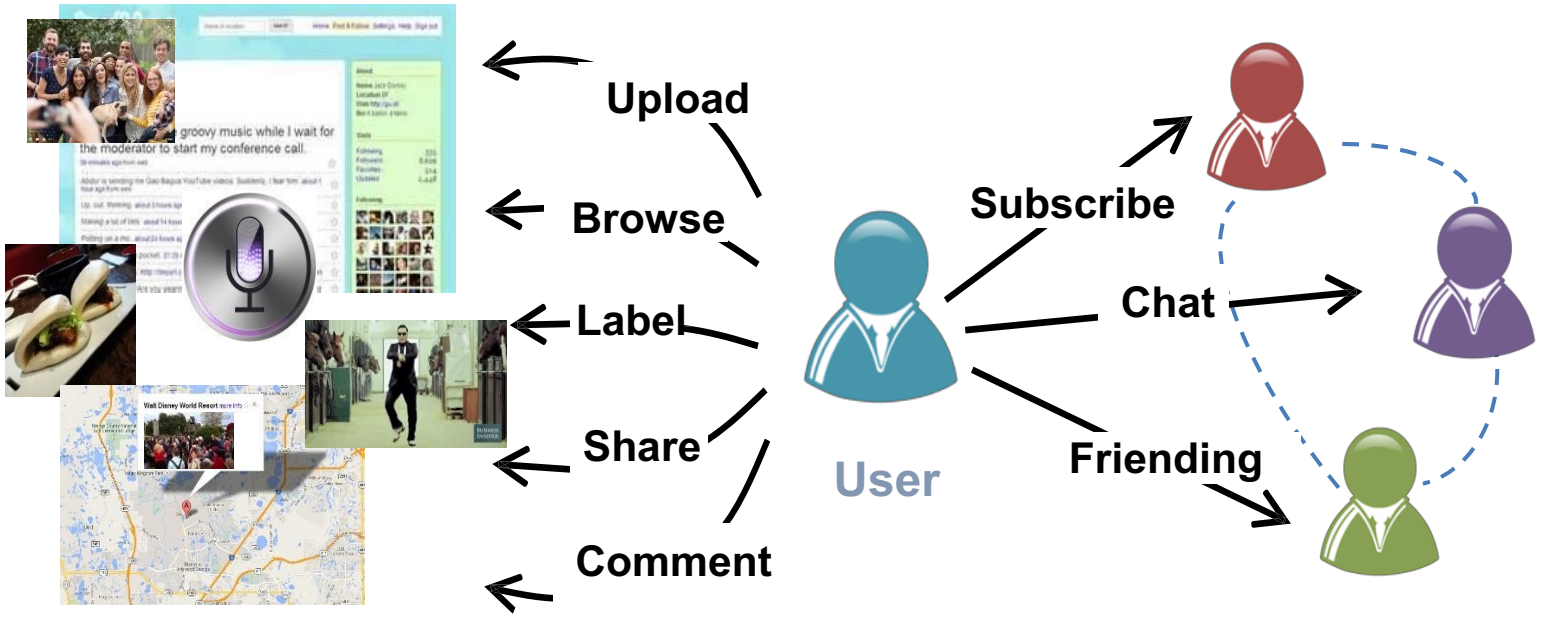
User Modeling: Analyze user behaviors to predict the unknown user features by constructing appropriate mathematical models for users' profiles/attributes/interests.



User Modeling

Social Media Context

User profiles, behaviors, time series, social relations, interaction network, etc.——Provide a wealth of source materials and bases for user modeling.



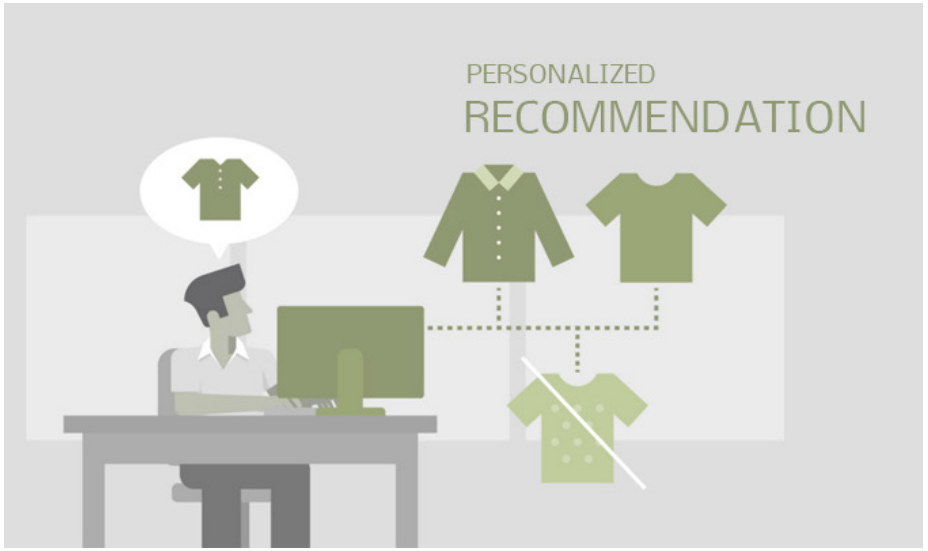
Social Media Behaviors

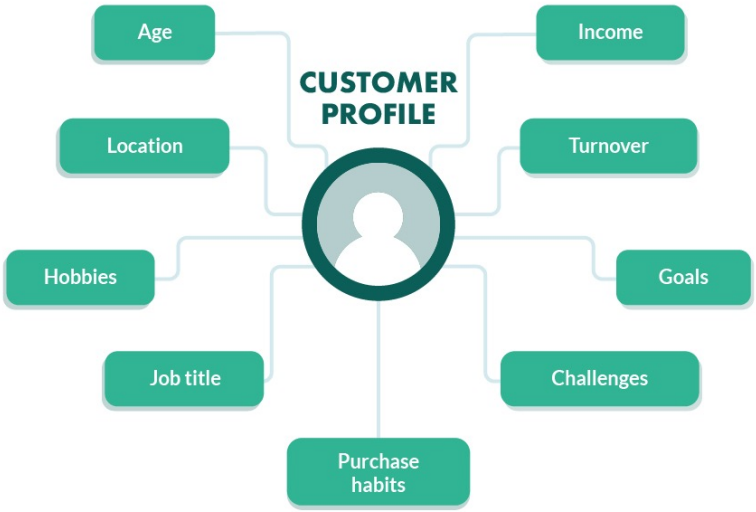
Social relations

- **Theoretical Significance**
  - Massive heterogeneous multi-modal data calculation
  - Dynamic sequence modeling
  - User modeling and understanding
- **Application Value**
  - Multimedia Retrieval
  - Recommender System
  - Accurate Marketing
  - Personalized Service（education、Social、entertainment, etc.）

User Modeling

*We take **recommender systems** as the example in this talk.*





Personalized Search

Recommender Systems √

Customer Profiles

Recommender System

A recommender system, or a recommendation system ,is a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item. The items can be products, news, movies, videos, friends, etc.

Recommendation has been widely applied in online services:

- E-commerce, Content Sharing, Social Networking ...

Recommender System

Effectiveness

- Accuracy
- CTR (Click Through Rate)
- Conversion rate
- Retention rate

Sufficiency

There are many application scenarios where considering **short-term user interests** and **longer-term sequential patterns** can be central to the success of a recommender[1].

Diversity

The more diverse the recommendations, the greater the chance of long-term retention.
——Dr. Huanhuan Cao from ByteDance.

Efficiency

In production environments, most recommender systems should compute predictions in real time[2]. Some researches[2, 3] have already focused on the efficiency problems while training and applying recommendation models.

[1] Quadrana M, Cremonesi P, Jannach D. Sequence-aware recommender systems[J]. ACM Computing Surveys (CSUR), 2018, 51(4): 1-36.
[2] Cacheda F, Carneiro V, Fernández D, et al. Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems[J]. ACM Transactions on the Web (TWEB), 2011, 5(1): 1-33.
[3] Huang X, Qian S, Fang Q, et al. Csan: Contextual self-attention network for user sequential recommendation[C]. Proceedings of the 26th ACM international conference on Multimedia. 2018: 447-455.

Trustworthy AI

Trustworthy

"worthy of trust of confidence; reliable, dependable" in the Oxford English Dictionary

Trustworthy Artificial Intelligence

Trustworthy AI as programs and systems built to solve problems like a human, which bring benefits and convenience to people with no threat or risk of harm.

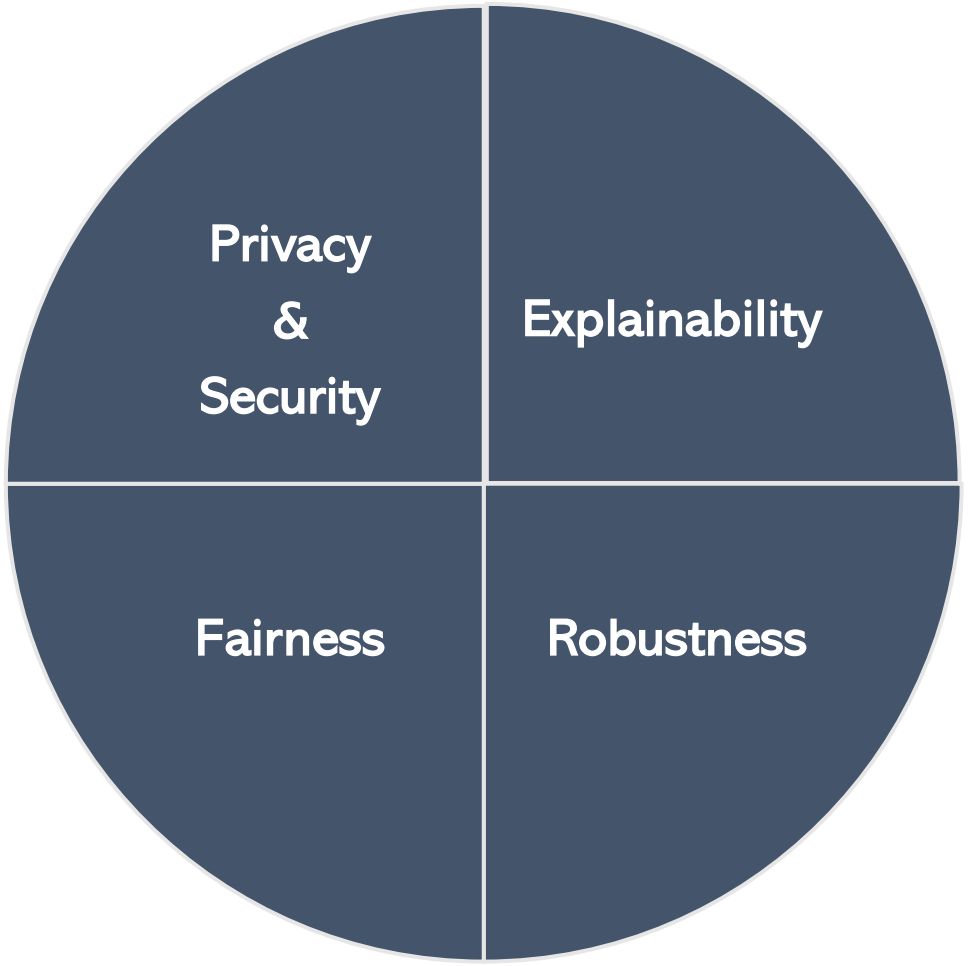
A summary of principles for Trustworthy AI from different perspectives.

Perspective	Principles
Technical	Accuracy, Robustness, Explainability
User	Availability, Usability, Safety, Privacy, Autonomy
Social	Law-abiding, Ethical, Fair, Accountable, Environmental-friendly

Six key dimensions of trustworthy AI.

Liu H, Wang Y, Fan W, et al. Trustworthy ai: A computational perspective[J]. arXiv preprint arXiv:2107.06641, 2021.

Trustworthy User Modeling



BEYOND PREDICTION ACCURACY

Explainability

- Suggests that the decision mechanism system should be able to be explained to stakeholders (who should be able to understand the explanation).

Robustness

- Requires the system to be robust to the noisy perturbations of inputs and to be able to make secure decisions.

Fairness

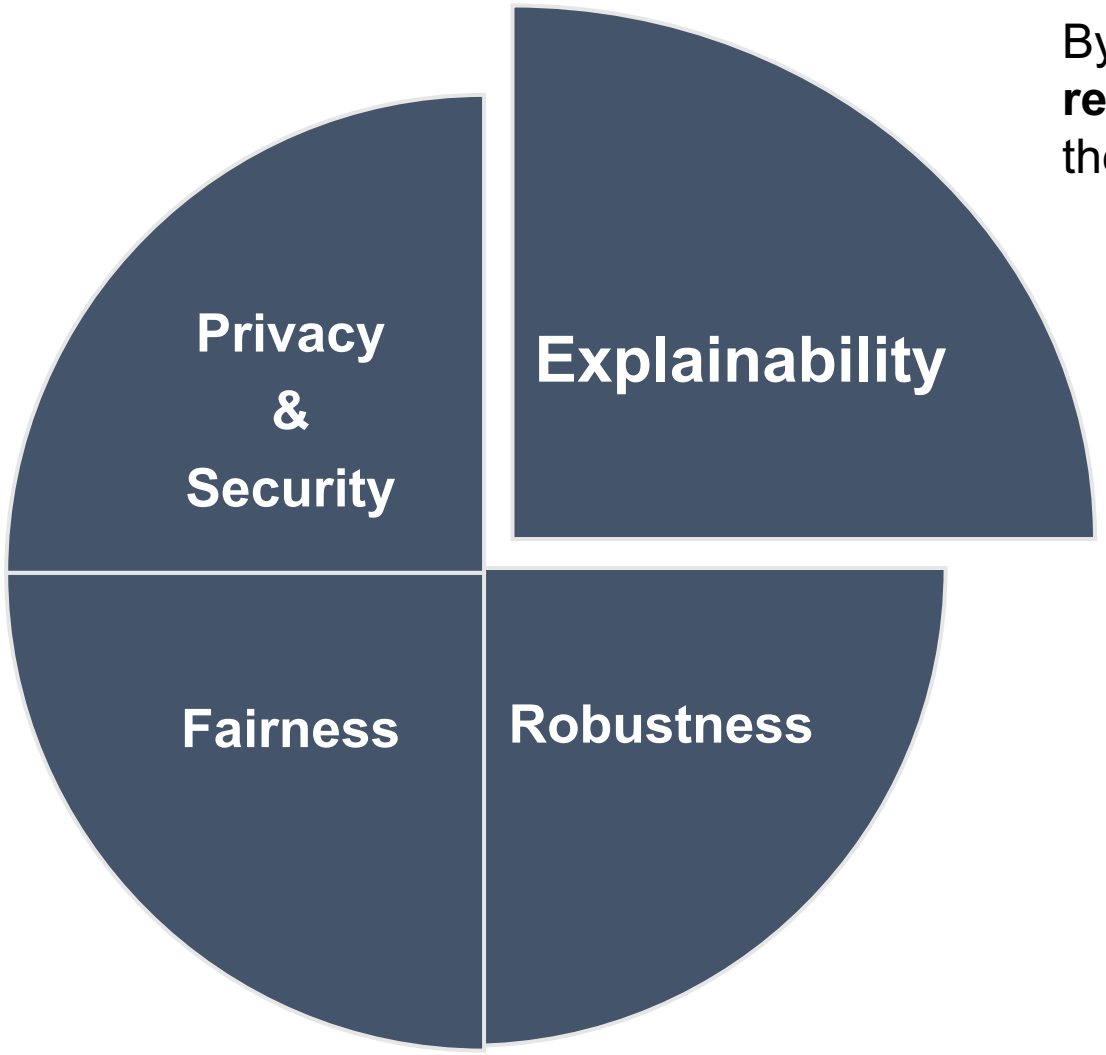
- It is expected to avoid unfair bias toward certain groups or individuals.

Privacy & Security

- Requires the system to avoid leaking any private information.

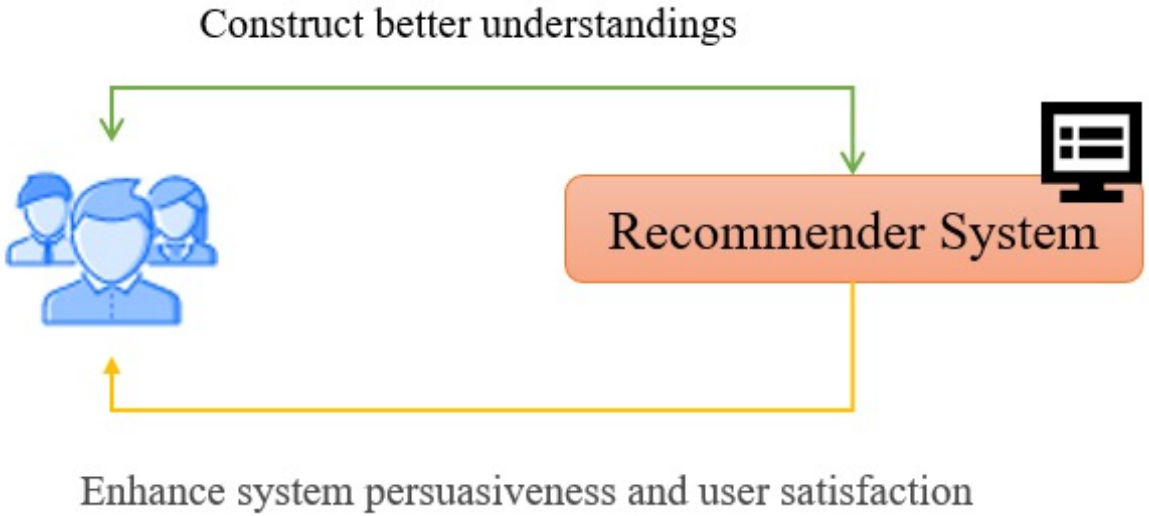
Liu H, Wang Y, Fan W, et al. Trustworthy ai: A computational perspective[J]. arXiv preprint arXiv:2107.06641, 2021.

Explainability



By explaining **how** the **system works** and/or **why** a **product is recommended**, the system becomes more transparent and has the potential to [1]:

- Allow users to tell when the system is wrong (scrutability)
- Increase users' confidence or trust in the system.
- Help users make better (effectiveness) and faster (efficiency) decisions.
- Convince users to try or buy (persuasiveness), or increase the ease of the user enjoyment (satisfaction).



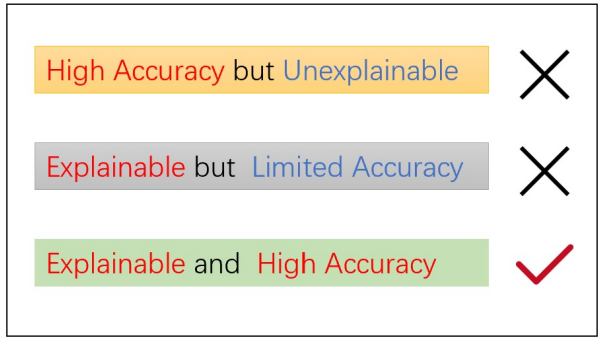
Zhang Y, Lai G, Zhang M, et al. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis[C]. Proceedings of the 37th international ACM SIGIR conference on Research &development in information retrieval. 2014: 83-92.

Explain for what

Deep Learning methods are **black-box** models but can reach considerable (high accuracy) results.

Intransparent and hard to understand

Models are facing **trade-off problems** between **accuracy** and **explainability**. **Explainable recommendations** should pursue **both** accuracy and explainability.



Explain for what?

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)?

@Min Zhang

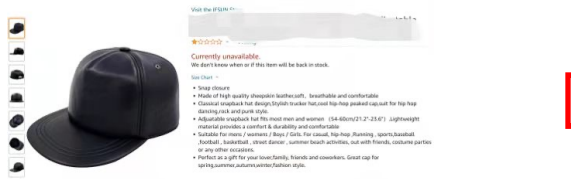
Gao J, Wang X, Wang Y, et al. Explainable recommendation through attentive multi-view learning[C]. Proceedings of the AAAI Conference on Artificial Intelligence. 2019, 33(01): 3622-3629.

Categorization: Representation Perspective

Representation Perspective

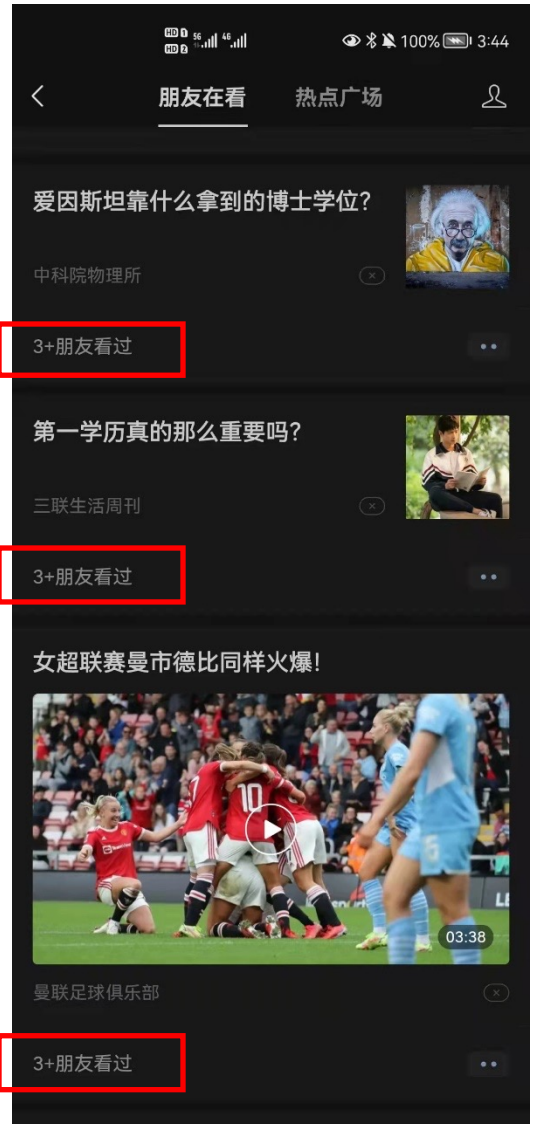
The information source or display style of the explanations, which represents the human-computer interaction(HCI) perspective of explainable recommendation research.

- Relevant User or Item Explanation
- Feature-based Explanation
- Opinion-based Explanation
- Sentence Explanation
- Visual Explanation
- Social Explanation



Customers who viewed this item also viewed

More than 3 friends have read this.



Zhang Y, Chen X. Explainable Recommendation: A Survey and New Perspectives[J]. Foundations and Trends® in Information Retrieval, 2020, 14(1): 1-101.

Categorization: Model Perspective

Model perspective

The model to generate such explanations, which represents the machine learning (ML) perspective of explainable recommendation research[1].

Factorization Models	[Zhang Y, Lai G, Zhang M, et al. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis, SIGIR 2014.] [Tao Y, Jia Y, Wang N, et al. The fact: Taming latent factor models for explainability with factorization trees, SIGIR 2019.]
Topic Modeling	[Tan Y, Zhang M, Liu Y, et al. Rating-boosted latent topics: Understanding users and items with ratings and reviews, IJCAI 2016.] [Ren Z, Liang S, Li P, et al. Social collaborative viewpoint regression with explainable recommendations, WSDM 2017.]
Graph-based Models	[Heckel R, Vlachos M, Parnell T, et al. Scalable and interpretable product recommendations via overlapping co-clustering, ICDE, 2017.] [Wang X, He X, Feng F, et al. Tem: Tree-enhanced embedding model for explainable recommendation, WWW 2018.]
Deep Learning	[Seo S, Huang J, Yang H, et al. Interpretable convolutional neural networks with dual local and global attention for review rating prediction, RecSys 2017.] [Gao J, Wang X, Wang Y, et al. Explainable recommendation through attentive multi-view learning, AAAI 2019.]
Model Agnostic and Post Hoc	[Wang X, Chen Y, Yang J, et al. A reinforcement learning framework for explainable recommendation, ICDM 2018.] [Peake G, Wang J. Explanation mining: Post hoc interpretability of latent factor models for recommendation systems, SIGKDD 2018.]
Rule Mining	[Balog K, Radlinski F, Arakelyan S. Transparent, scrutable and explainable user models for personalized recommendation, SIGIR 2019.]
Knowledge Graph-based	[Huang X, Fang Q, Qian S, et al. Explainable interaction-driven user modeling over knowledge graph for sequential recommendation, ACM MM 2019.] [Wang X, Wang D, Xu C, et al. Explainable reasoning over knowledge graphs for recommendation, AAAI 2019.]

Zhang Y, Chen X. Explainable recommendation: A survey and new perspectives[J]. Foundations and Trends in Information Retrieval, 2020, 14(1): 1-101.

Evaluations for Explainability

Conventional Method: MAE and RMSE for rating prediction task; Precision, recall, F-measure, NDCG for Top-N recommendation task.

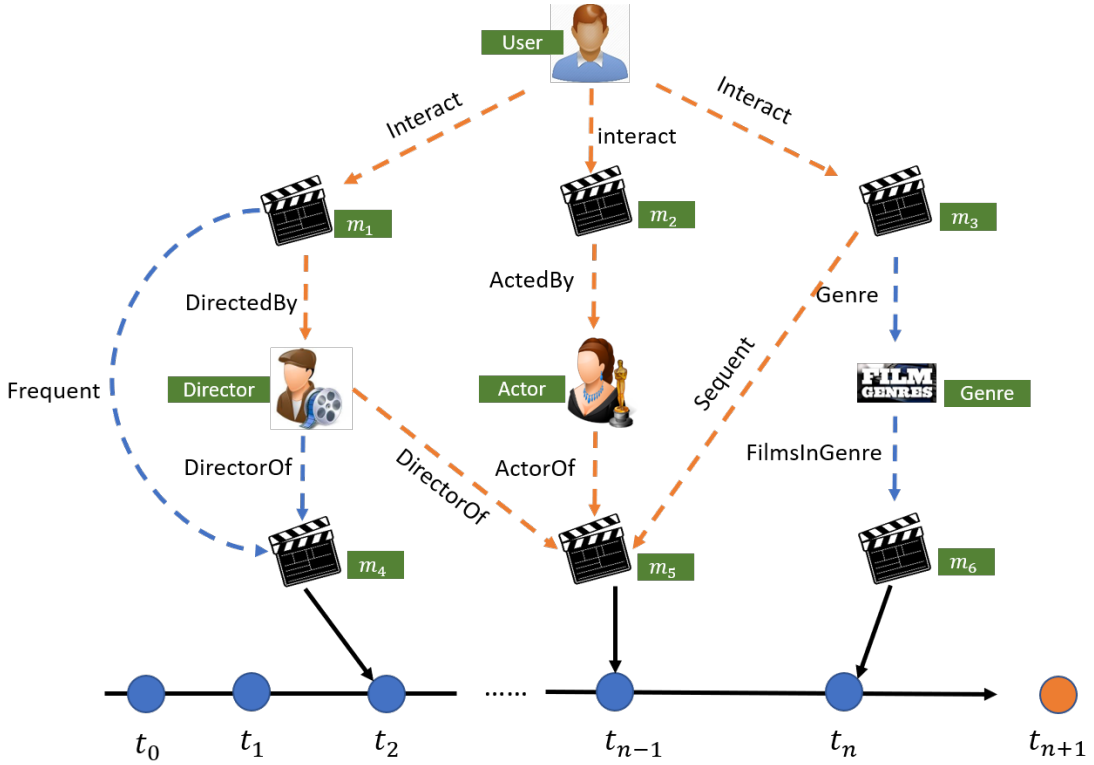
Some evaluations for explainability[1]:

- **User study**: To perform some surveys about the explanations generated by recommender systems on hired users or volunteers. The survey can ask users to evaluate[2] or gather useful information from users during their interactions with the recommender systems.
- **Online evaluation** such as to **conduct A/B test**. Research may focus on the click-through-rate(CTR), conversion rate, etc.
- **Offline evaluation**. Abdollahi and Nasraoui[3] proposed Explainability Precision(EP) and Expalainability Recall(ER). Some approaches focus on the quality of the explanations. For example, we can apply text-based measures for explanation sentences.
- **Case study** which can help to understand the intuition behind the explainable recommendation model and the effectiveness of explanations.

[1] Zhang Y, Chen X. Explainable recommendation: A survey and new perspectives[J]. Foundations and Trends in Information Retrieval, 2020, 14(1): 1-101.
[2] Wang N, Wang H, Jia Y, et al. Explainable recommendation via multi-task learning in opinionated text data[C]. The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. 2018: 165-174.
[3] Abdollahi B, Nasraoui O. Explainable matrix factorization for collaborative filtering[C]. Proceedings of the 25th International Conference Companion on World Wide Web. 2016: 5-6.

Knowledge graph-based method: EIUM

- **Task**: Explainable sequential recommendation.
- **Motivation**: Since the interest of the user is evolving dynamically, how to capture the user's dynamics interests accurately and explainably is the focus of our work.
- **Challenges**: Most of the deep-learning-based methods do not consider providing users with credible explanations while recommending. KG-based methods do not consider the chronology of user behaviors, resulting in the difficulties of modeling users' dynamic preferences with time drifting.



Huang X, Fang Q, Qian S, et al. Explainable interaction-driven user modeling over knowledge graph for sequential recommendation[C]//Proceedings of the 27th ACM International Conference on Multimedia. 2019: 548-556.

Methodology: Multi-model Fusion

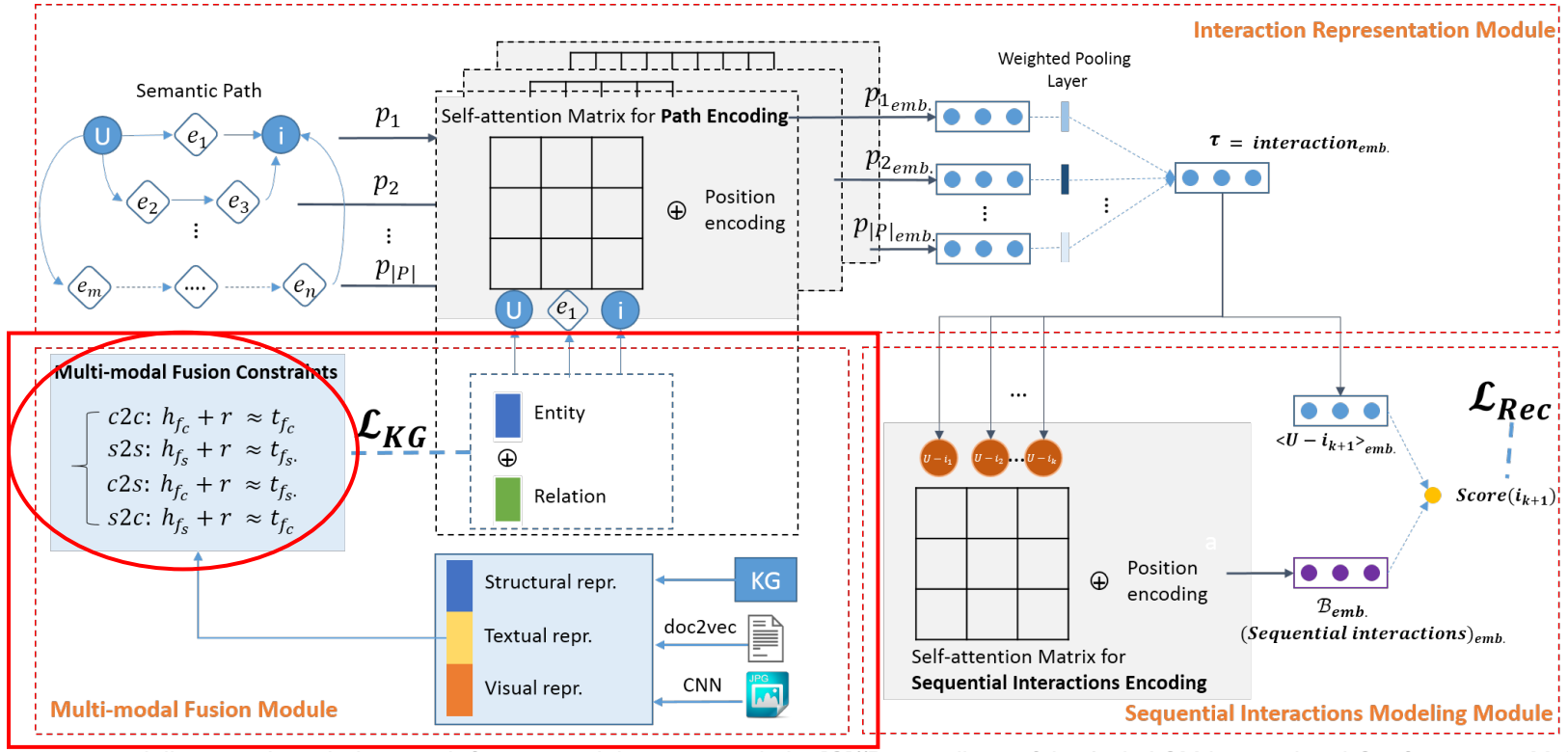
Employing **multi-modal fusion** which benefits from the structural constraints in KG, where three kinds of modalities are involved for better representation learning.

- Content features
  - Textural features
  - Visual features
- Structural features： f_s

$$item^{textual} \in \mathcal{R}^{300} = \frac{1}{n}\sum_{i=1}^{n} w_i$$

$$f_c = \sigma(W_f[item^{textual}, item^{visual}] + b_f)$$

$$item^{visual} \in \mathcal{R}^{300}$$

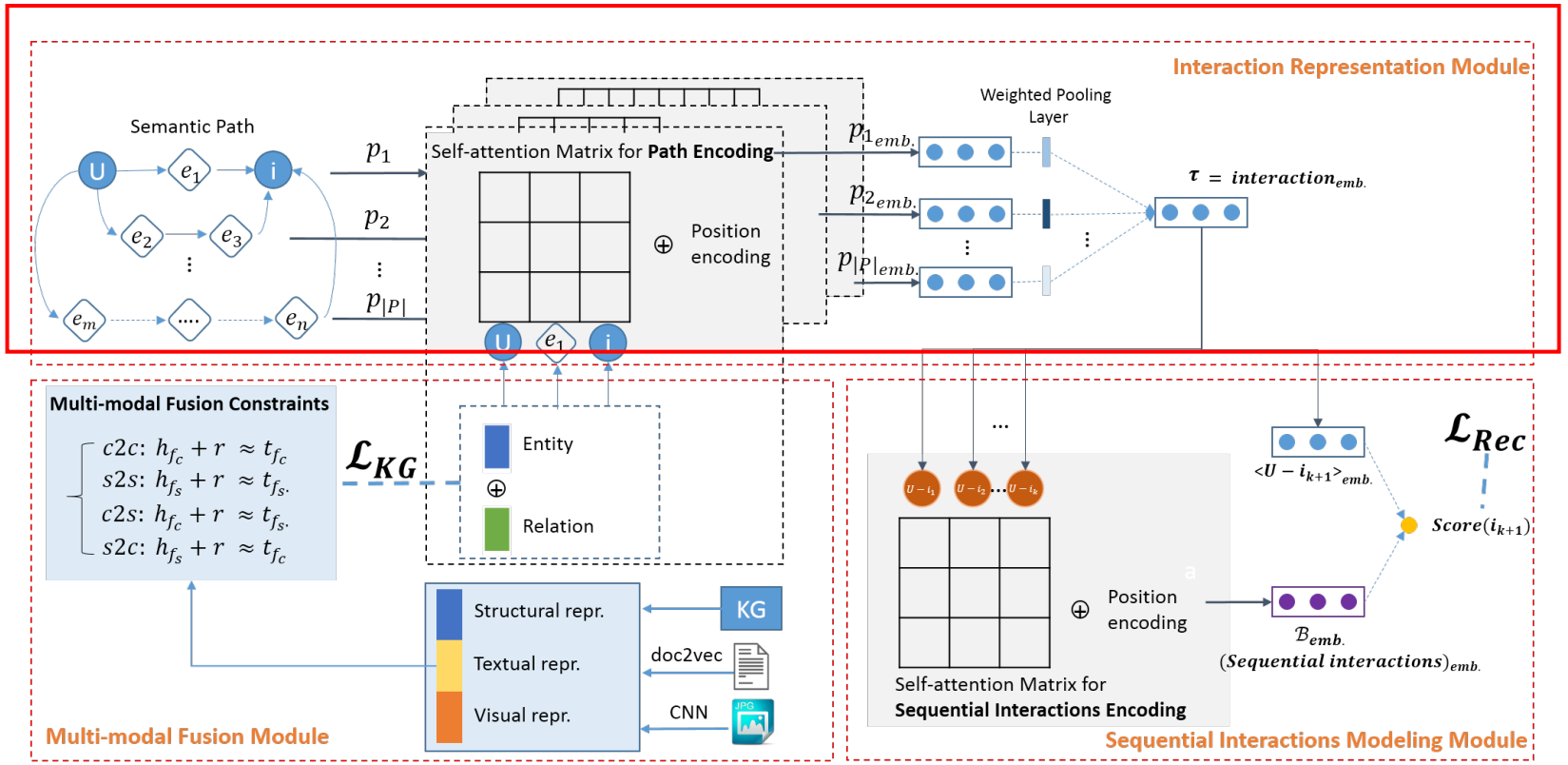


Huang X, Fang Q, Qian S, et al. Explainable interaction-driven user modeling over knowledge graph for sequential recommendation[C]//Proceedings of the 27th ACM International Conference on Multimedia. 2019: 548-556.

Methodology: Interaction Representation

- Paths P(*u*,*i*) between a user-item pair contains different semantic information about the interaction between the user and the item. Therefore, the **semantic representation of the interaction** can be obtained by modeling the corresponding paths.

$$p_{l_{emb.}} = mean\text{-}pooling(o_j)_{j=1}^{L} = \frac{1}{L}\sum_{j=1}^{L} o_j$$



- We use a weighted pooling layer to help distinguish the path importance. An attention mechanism is adopted to address this issue.

$$query = \sigma_q(W_q[u,i] + b_q)$$

$$w(\mathcal{P}(u,i)) = [w_1, w_2, ..., w_{|\mathcal{P}|}]$$

- The unified **interaction representation** is obtained by aggregating the weighted paths.

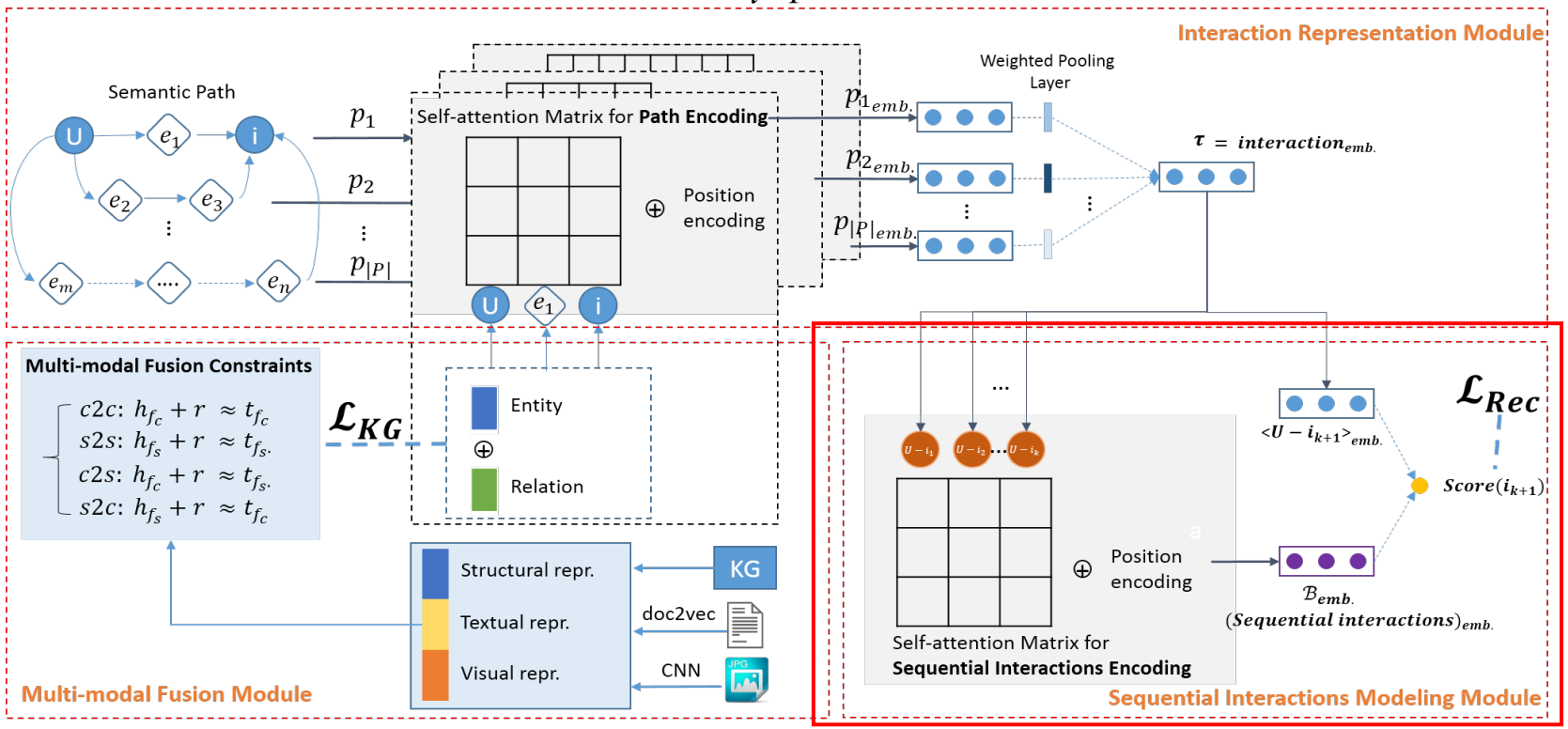
$$\tau = interaction_{emb.} = \sum_{l=1}^{|\mathcal{P}|} w_l \cdot p_{l_{emb.}}$$

Huang X, Fang Q, Qian S, et al. Explainable interaction-driven user modeling over knowledge graph for sequential recommendation[C]//Proceedings of the 27th ACM International Conference on Multimedia. 2019: 548-556.

Methodology: Sequential Interactions Modeling

- A user's historical records are a sequence in chronological order, thus her/his subsequent item can be predicted by SR methods. Given a user's interacted item sequence $\{i_t, t = 1, 2, ..., T\}$, the sequential interactions can be represented as $\mathcal{B} = \{\tau_t, t = 1, 2, ..., T\}$, the **user preference representation** based on sequential interactions is defined as:

$$\mathcal{B}_{emb.} = \sum_{t=1}^{T} w_t \cdot e_{\tau_t}$$



Huang X, Fang Q, Qian S, et al. Explainable interaction-driven user modeling over knowledge graph for sequential recommendation[C]//Proceedings of the 27th ACM International Conference on Multimedia. 2019: 548-556.

Joint Learning

- Recommendation Loss：

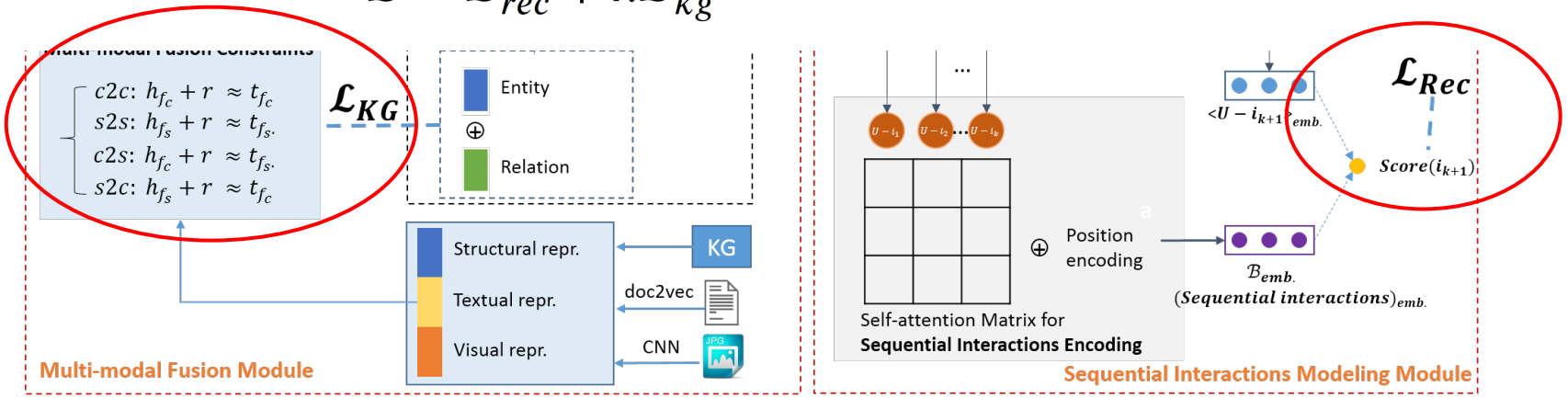
$$\mathcal{L}_{rec} = \sum_{u} \sum_{v} \sum_{v' \in \hat{\mathcal{V}}} -log\sigma[D(uv) - D(uv')]$$

- Multi-modal fusion constrains：

$$\mathcal{L}_{kg} = \mathcal{L}_{c2c} + \mathcal{L}_{s2s} + \mathcal{L}_{c2s} + \mathcal{L}_{s2c}$$
$$= \frac{1}{4} \sum_{i} ||h + r - t||, i \in \{c2c, s2s, c2s, s2c\}$$

- Minimizing the overall objective function that jointly evaluates the performance：

$$\mathcal{L} = \mathcal{L}_{rec} + \lambda \mathcal{L}_{kg}$$



Huang X, Fang Q, Qian S, et al. Explainable interaction-driven user modeling over knowledge graph for sequential recommendation[C]//Proceedings of the 27th ACM International Conference on Multimedia. 2019: 548-556.

Experimental Settings

Datasets

User-Item Interaction	# Users	138,287
	# Items	13,047
	# Interactions	9,971,069
	# Avg.interaction	72
Knowledge Graph	# Entities	40,529
	# Relations	66
	# Triplets	7,422,000
Path	# Paths	146,222,314
	# Avg.path	16.62
	# Avg.path_length	5.04
Train-test Dataset	# Train samples	691,435
	# Test samples	138,287
	# Images	13047

Comparison Methods

- **BPR**. Bayesian personalized ranking[16] is a pairwise ranking framework which takes Matrix Factorization as the underlying predictor.
- **Bi-LSTM**. Bi-directional Long short-term memory (LSTM) units are building units for the layer of RNN, which are used to capture sequential dependencies and make predictions[31].
- **Bi-LSTM with attention**. Incorporating attention mechanism into Bi-LSTM methods mentioned above.
- **ATRank**. ATRank[33] is an attention-based user modeling framework which encoding sequential behaviors based only on the self-attention mechanism.
- **CKE**. CKE[29] is a recently proposed state-of-the-art method that incorporates KG embedding to improve the recommendation performance.
- **KTUP**. KTUP[3] is a knowledge-enhanced translation-based user preference model. It transfers the relation embeddings as well as entity embeddings learned from KG to the user preference model and simultaneously training two different tasks.
- **EIUM**. It is our proposed model detailed in Section 4.

Huang X, Fang Q, Qian S, et al. Explainable interaction-driven user modeling over knowledge graph for sequential recommendation[C]//Proceedings of the 27th ACM International Conference on Multimedia. 2019: 548-556.

Effectiveness and Explainability

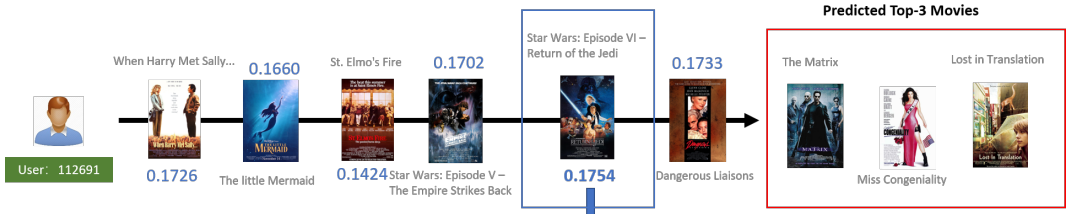
Effectiveness

- Capturing the **interaction-level** user dynamic preference, which is a high-level representation.
- Incorporating Knowledge Graph endows the recommendation system the ability of **path-wise** explanation.

Datasets	Methods	Evaluation Metrics					
		AUC	MAP	Hit@5	Hit@10	NDCG@5	NDCG@10
MovieLens-20M + Freebase	BPR	0.9065 (-0.18%)	0.3312 (-21.05%)	0.4920 (-19.65%)	0.6653 (-16.68%)	0.3578 (-21.95%)	0.3990 (-21.19%)
	Bi-LSTM	0.8800 (-3.09%)	0.3278 (-21.86%)	0.5583 (-8.82%)	0.7180 (-10.08%)	0.3749 (-18.22%)	0.4147 (-18.09%)
	Bi-LSTM+att.	0.8897 (-2.03%)	0.3606 (-14.04%)	0.5944* (-2.92%)	0.7409* (-7.21%)	0.4095 (-10.67%)	0.4460 (-11.91%)
	ATRank	0.8724 (-3.93%)	0.3454 (-17.66%)	0.5561 (-9.18%)	0.7057 (-11.62%)	0.3877 (-15.42%)	0.4250 (-16.06%)
	CKE	0.9054 (-0.30%)	0.3869* (-7.77%)	0.5790 (-5.44%)	0.7175 (-10.14%)	0.4261* (-7.05%)	0.4571* (-9.72%)
	KTUP	**0.9187*** (+1.17%)	0.3829 (-8.72%)	0.5662 (-7.53%)	0.7085 (-11.27%)	0.4196 (-8.46%)	0.4516 (-10.80%)
	EIUM	0.9081	**0.4195**	**0.6123**	**0.7985**	**0.4584**	**0.5063**

Explainability

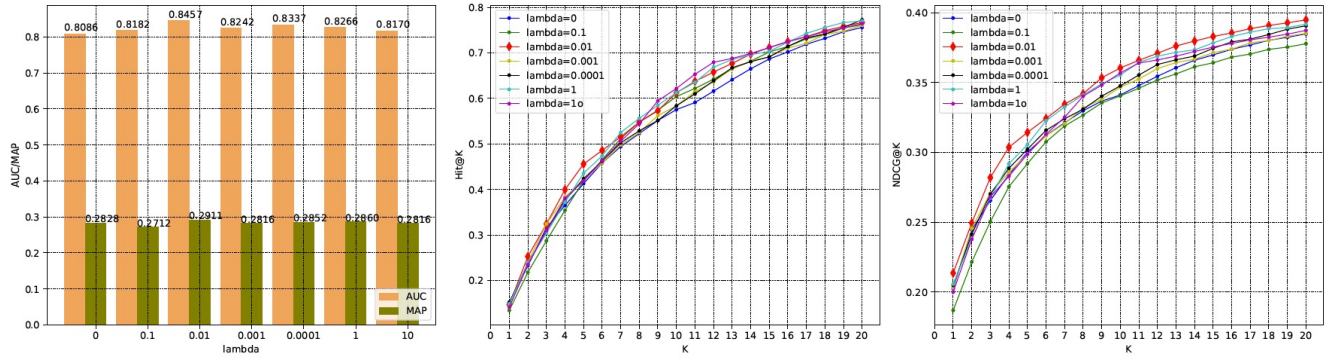
Interaction-wise Explanation



Path-wise Explanation

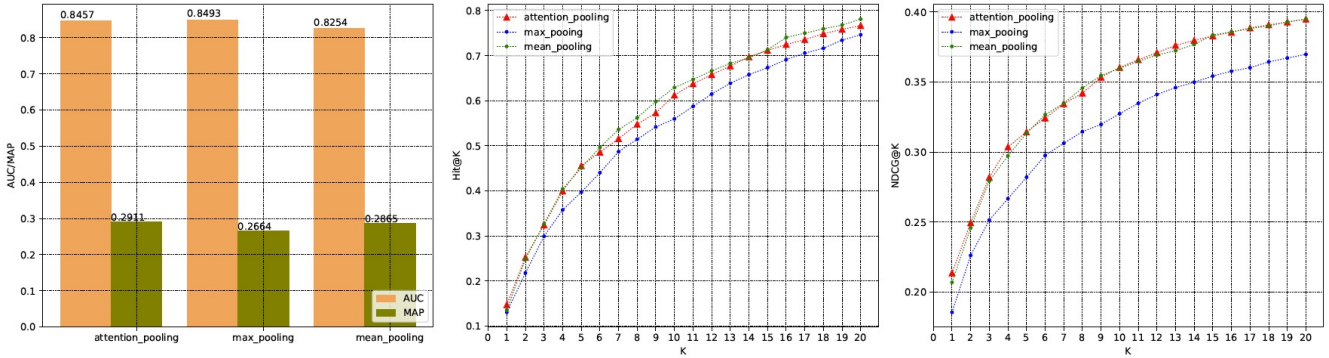
Huang X, Fang Q, Qian S, et al. Explainable interaction-driven user modeling over knowledge graph for sequential recommendation[C]//Proceedings of the 27th ACM International Conference on Multimedia. 2019: 548-556.

Parameter Analysis



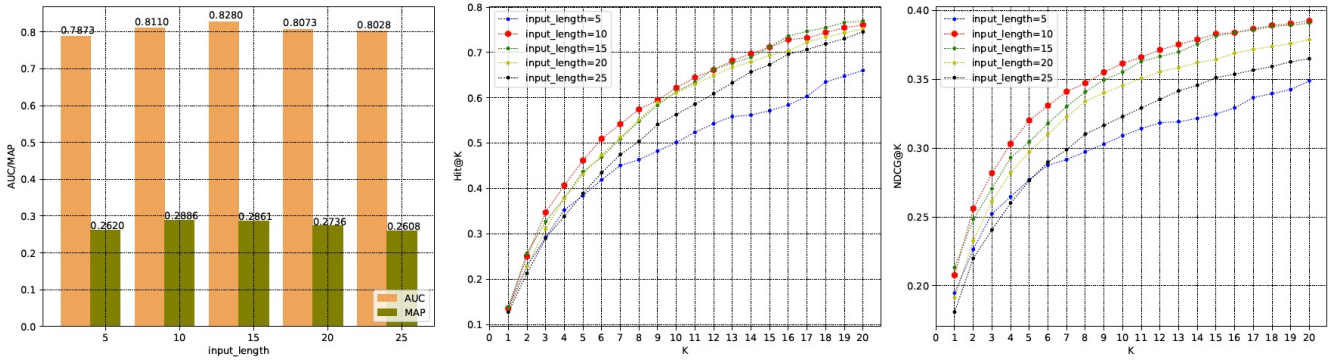
- Effects of KG loss

  the accuracy is not monotone with λ, since if λ is too large, the recommendation loss will be invalid, and if the λ is too small, the graph structure information constraints will not work for making full use of structural knowledge from KG. The EIUM achieves the best performance when λ = 0.01.



- Effects of pooling manner

  Using attention makes the comparable performance in the recommendation. In addition, the interactive paths can be selected according to the attention score which also can provide a more intuitive explanation for users.

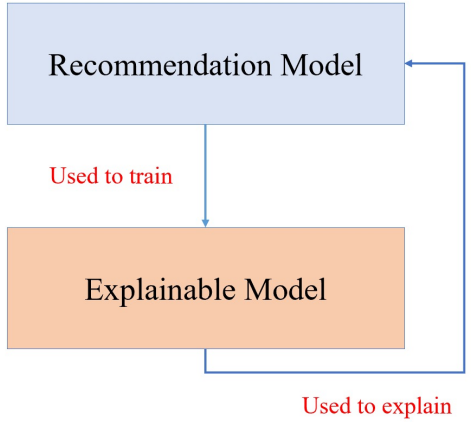


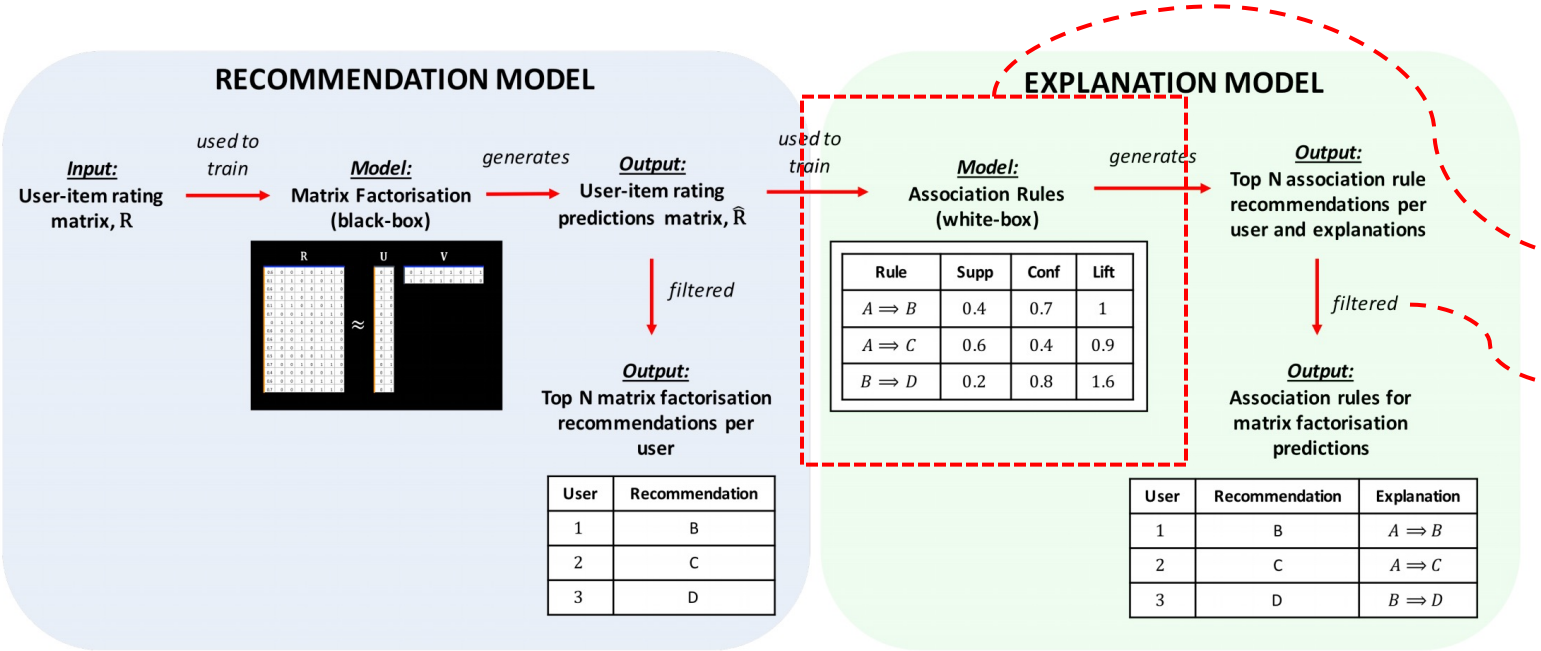
- Effects of length of historical behaviors

  Many behaviors with long intervals may introduce noise information into the system. It demonstrates the rationality of our hypothesis, and the appropriate input-length is 10.

Post Hoc Method

- **Task:** Apply a novel post hoc interpretability approach to a latent factor model for recommendation systems.
- **Motivation**:
  - Many recommendation systems use black-box latent factor models that provide no explanation of why a recommendation has been made.
  - Post hoc interpretability makes no changes to the black-box model itself, predictive performance can be **maintained**.





Algorithm 1 Approximate matrix factorisation using global association rules

Input: Matrix factorisation predictions $\hat{R}$; training data

1: For each user i, generate a transaction list T_i of the index of top D matrix factorisation predictions, $\hat{R}_i$
2: Generate the set $\mathcal{Z}_i$ of rules ($X \Rightarrow Y$) that satisfy *min_supp*, *min_conf*, *min_lift* criteria from all transactions T. ▷ Rules generated by the apriori algorithm using the *apyori* [27] Python package
3: **for all** users, $i = 1 \ldots N$ **do**
4: Compute the list {unseen} of items Y where $X \Rightarrow Y$ if $X \in$ {train} and $Y \notin$ {train}.
5: Order {unseen} by *supp/conf/lift*. Compute {recommended} = {unseen}[: *top_n*]
6: Return list of recommended items, {recommended} and corresponding rules $X \Rightarrow Y$ as explanations.
7: **end for**

Output: {unseen}

Peake G, Wang J. Explanation mining: Post hoc interpretability of latent factor models for recommendation systems[C]. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018: 2060-2069.

Explainability Evaluation

Recommendation Accuracy for recommender: Boolean retrieval metrics **precision** and **recall** to evaluate the model's predictive performance on an unseen test set.

Model Fidelity for explanation model:

$$\text{Model Fidelity} = \frac{|\text{MF recommended items} \cap \text{AR retrieved items}|}{|\text{MF recommended items}|}$$

Rules	# clusters	Interestingness	Model Fidelity
Cluster	3000	Support	0.842794
		Confidence	0.852559
		Lift	0.66025
	700	Support	0.767592
		Confidence	0.799768
		Lift	0.460257
	10	Support	0.598649
		Confidence	0.640951
		Lift	0.423059

→ The explanation model performs better while using local association rules based on 3000 clusters of similar users in latent space.

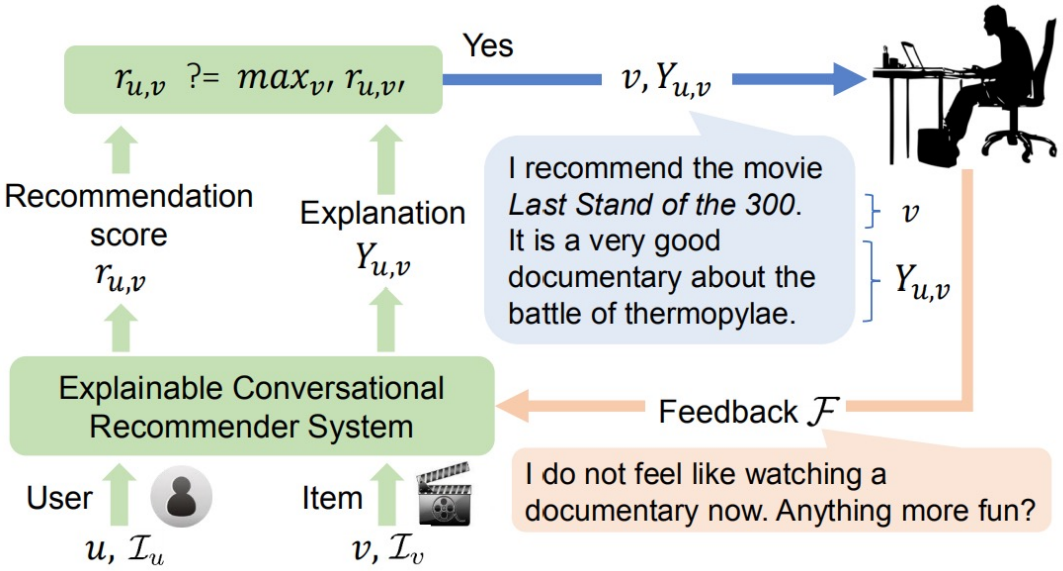
Peake G, Wang J. Explanation mining: Post hoc interpretability of latent factor models for recommendation systems[C]. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018: 2060-2069.

Sentence Explanation Level

- **Task**: Introduce explainable conversational recommendation, which enables incremental improvement of both recommendation accuracy and explanation quality through multi-turn user model conversation.
- **Motivation**: Prior works on conversational search and recommendation lack the capability for providing explanations.

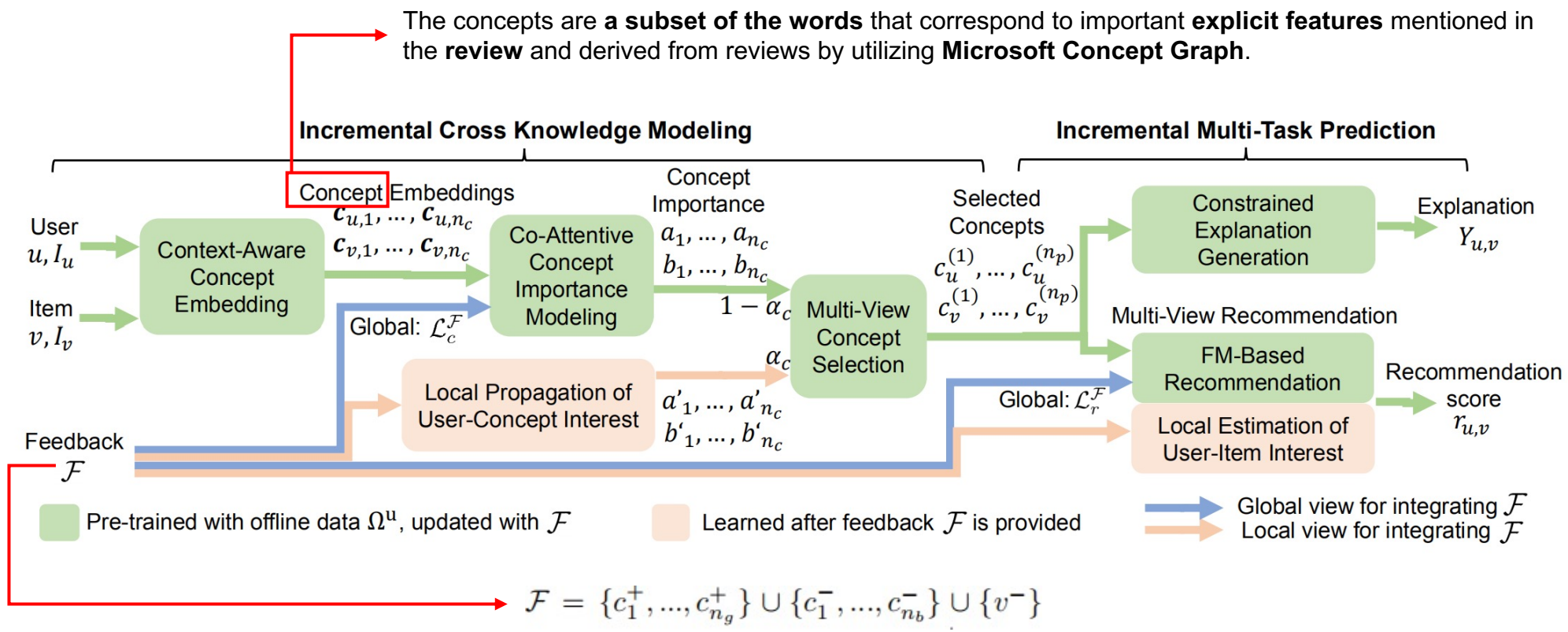
Model: I recommend Pulp Fiction. This is a dark comedy with a great cast.
User: I don't want to watch a comedy right now.
Model: How about Ice Age? It is a very good anime with a lot of action adventure.
User: I don't like anime, but action movie sounds good.
Model: I recommend Mission Impossible. This is by far the best of the action series.
User: Sounds great. Thanks for the recommendation!

Predefined Template | Recommended Item | Generated Explanation



Chen Z, Wang X, Xie X, et al. Towards Explainable Conversational Recommendation[C]. IJCAI. 2020: 2994-3000.

Methodology



Pre-trained model Loss:

$$\mathcal{L}^{\Omega} = \sum_{u \in U} (\mathcal{L}_r^{\Omega_u} + \lambda_n \mathcal{L}_n^{\Omega_u} + \lambda_c \mathcal{L}_c^{\Omega_u}) + \lambda_\theta \|\Theta\|_2^2$$

Global BPR loss

Generation loss under hard constraint

Generation loss under soft constraint(Concept relevance loss)

When a feedback is provided, incrementally update by minimizing:

$$\mathcal{L}^{\mathcal{F}} = \mathcal{L}_c^{\mathcal{F}} + \lambda_d \mathcal{L}_d^{\mathcal{F}} + \lambda_r \mathcal{L}_r^{\mathcal{F}} + \lambda_s \mathcal{L}_s^{\mathcal{F}} + \lambda'_\theta \|\Theta - \Theta'\|_2^2$$

Concept-level feedback loss

Multi-view Concept-level feedback loss

BPR loss with feedback

Multi-view BPR loss

Experimental Results

- **Recommendation Accuracy for recommender**: **HR** (hit ratio), **NDCG** (normalized discounted cumulative gain), and **MRR** (mean reciprocal rank).
- **Explanation Evaluation**: Adopt two measures **BLEU** and **ROUGE-L**, which are widely adopted to measure the similarity between ground truth and generated texts. Criterion **CSR** aims to measure the Concept-level feedback Satisfaction Ratio.

Dataset		Electronics				Movie&TV				Yelp			
Metric		NRT	CAML	Ours-G	Ours	NRT	CAML	Ours-G	Ours	NRT	CAML	Ours-G	Ours
O1	HR	0.176	0.202	0.272	**0.448**	0.194	0.272	0.274	**0.386**	0.204	0.234	0.258	**0.432**
	NDCG	0.285	0.311	0.312	**0.534**	0.309	0.314	0.354	**0.474**	0.310	0.335	0.343	**0.510**
	MRR	0.244	0.279	0.303	**0.500**	0.266	0.285	0.427	**0.439**	0.272	0.298	0.313	**0.479**
O2	BLEU	1.04	1.12	1.49	**2.10**	1.49	1.47	1.53	**2.31**	1.26	1.15	1.52	**2.17**
	ROUGE-L	13.16	14.74	15.73	**19.88**	14.43	14.56	15.78	**18.92**	11.36	12.00	13.37	**19.34**
O3	CSR	0.08	0.12	0.38	**0.94**	0.20	0.19	0.45	**0.97**	0.14	0.13	0.39	**0.97**

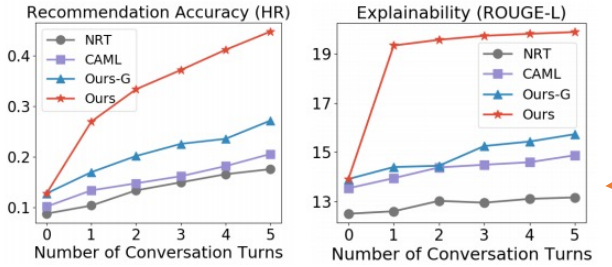


Figure 4: Recommendation accuracy and explainability on Electronics at different conversation turns.

Performance gain during conversation and Human evaluation

As the number of conversation turns increases, the number of feedbacks increases, and the performance of all methods improves.

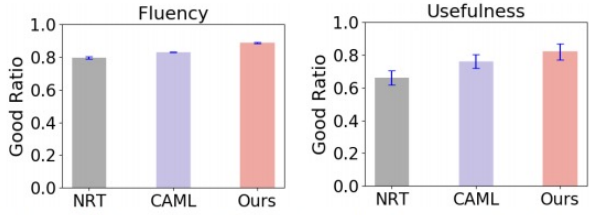


Figure 5: Human evaluation of explanation fluency and usefulness.

This research hires three experienced human assessors to label the explanations generated after 5 turns of conversation. 100 test cases are sampled from the Electronics dataset, and each assessor labels whether an explanation is fluent and whether it is useful.

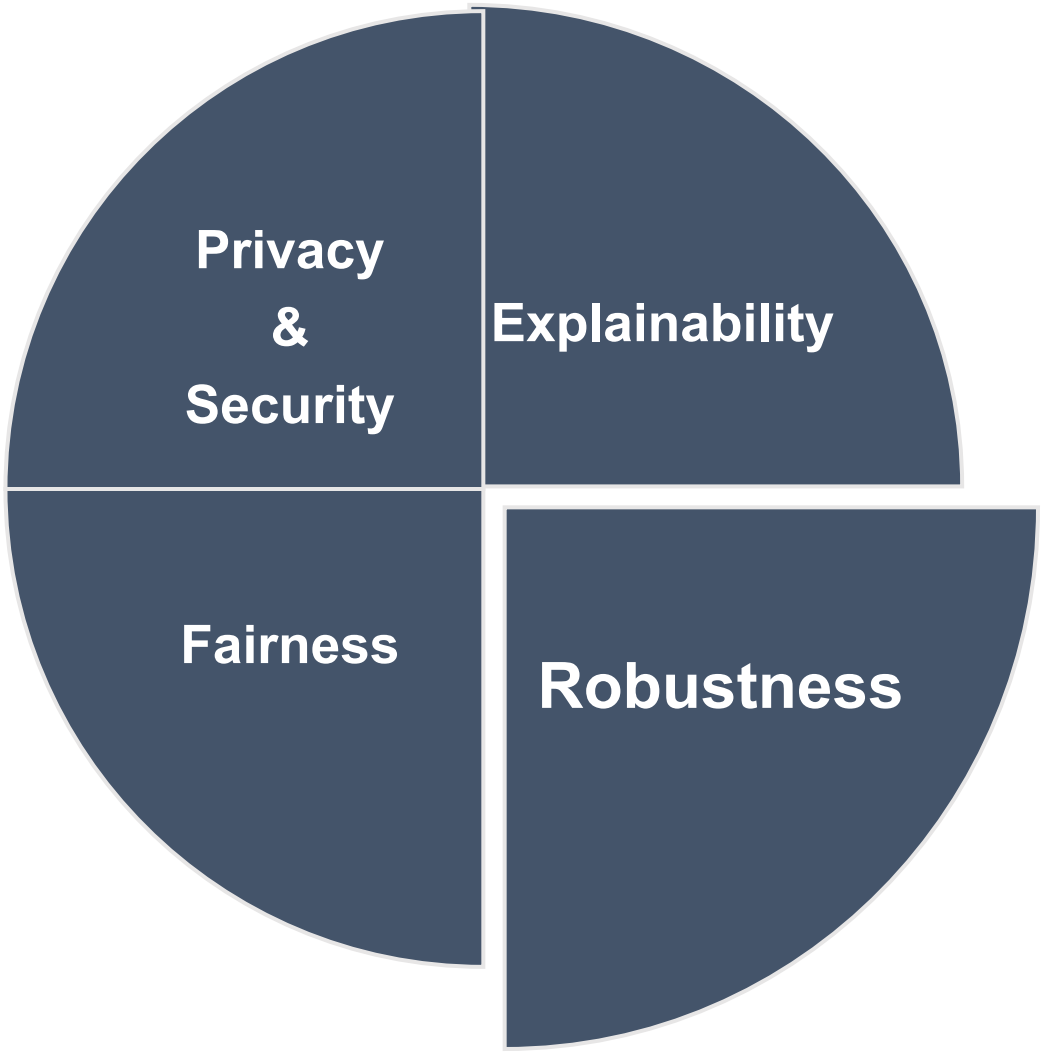
Future Work

- Evaluation Method.

  Except for online A/B tests or user studies, we should find an efficient offline method or build a benchmark.
- Combined with other famous deep learning mechanisms and Knowledge Graphs.
  - Follow with advanced models in CV, NLP, etc.
  - Use knowledge graph as external knowledge to enhance explainability.
- Conversational Recommendation.
- Causal inference
  - Xu and Li[1] propose to construct causal explainable recommendation which aims to provide post-hoc explanations for the recommendations.
  - A work takes the insights of counterfactual reasoning from causal inference for explainable recommendation has been accepted by CIKM 2021[2].

[1] Xu S, Li Y, Liu S, et al. Learning Causal Explanations for Recommendation[C]. The 1st International Workshop on Causality in Search and Recommendation. 2021.
[2] Tan J, Xu S, Ge Y, et al. Counterfactual Explainable Recommendation[C]. CIKM'21: Proceedings of the 30th ACM International Conference on Information and Knowledge Management. 2021..

Robustness



Robustness

- Requires the system to be robust to the noisy perturbations of inputs and to be able to make secure decisions.

Robustness Issues

Internal affair

- Data: Noise, unbalanced data, missing data
- Model: Parameter sensitivity

External affair

- Attack and Defense
  - Data poisoning/shilling attacks: promote/demote a set of items

Robustness

Robustness Issues

Internal affair

- Data: Noise, unbalanced data, missing data
- Model: Parameter sensitivity

External affair

- Attack and Defense
  - Data poisoning/shilling attacks: promote/demote a set of items

One of the missing data problems in recommender systems:

New users/items

→ **Cold-start problem**

@Min Zhang

Active user modeling

→ **Conversational recommendation**

- to increase the robustness of recommendation models with the existence of shilling attacks
- to detect and block fraudsters from the data.

Robustness

Robustness Issues

Internal affair

- Data: Noise, unbalanced data, missing data
- Model: Parameter sensitivity

External affair

- Attack and Defense
  - Data poisoning/shilling attacks: promote/demote a set of items

One of the missing data problems in recommender systems:

New users/items

→ **Cold-start problem**

@Min Zhang

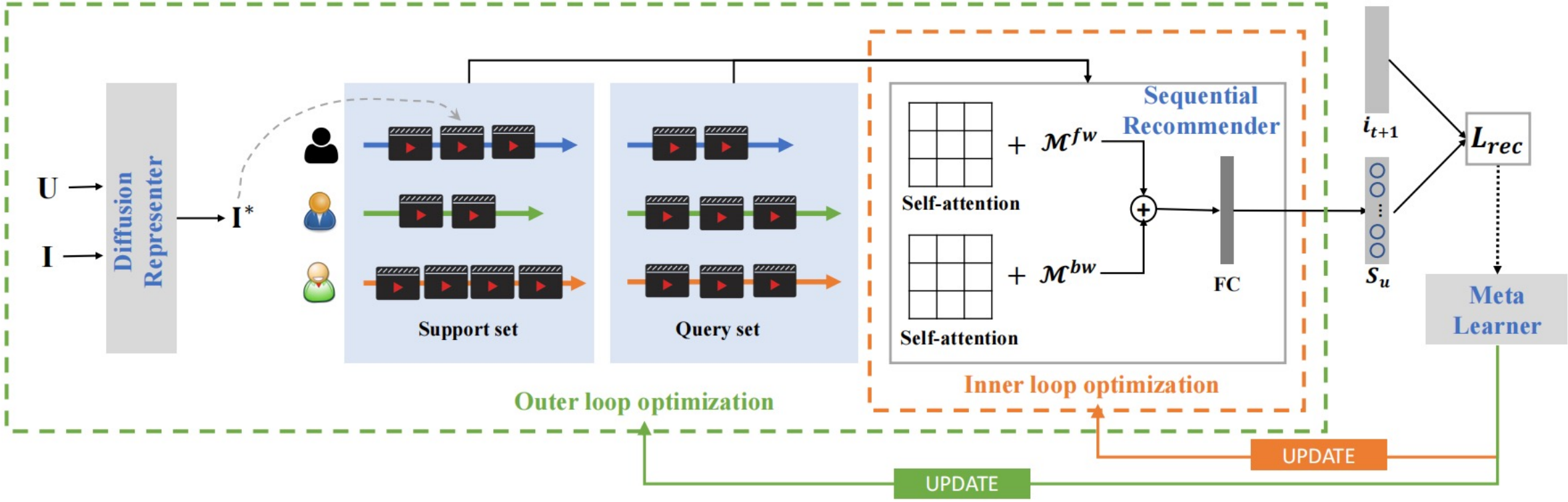
Cold-start problem

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

- Investigating more information
  - User profile
  - Item context
  - Knowledge graph
- Cross Domain/Platform
  - Social network
  - Heterogeneous behaviors
- Meta learning

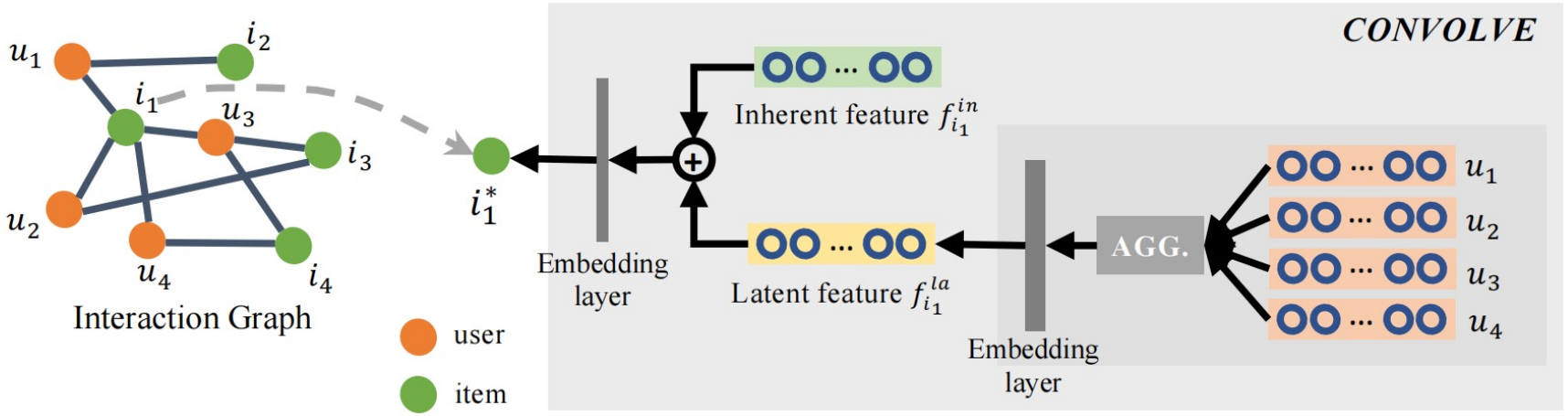
metaCSR: meta-learning based Cold-start Sequential Recommendation framework

- We focus on the cold-start sequential recommendation task where common patterns of sequential behaviors are mined and learning through our meta-learning based algorithm.
- Our proposed metaCSR is a general framework for CSR, which does not require any additional side information other than user ID, item ID, and interaction matrix of users on items, and can still achieve good results on the CSR task.



Huang X, Sang J, Xu C. Learning to Learn a Sequential Recommender[J]. IEEE Transactions on Information Systems (TOIS), 2021, Accepted Arxiv Link: https://arxiv.org/abs/2110.09083.

metaCSR: Diffusion Representer



Algorithm 1: CONVOLVE

Input: Inherent feature embedding f_v^{in} of entity v; A set of neighbor representation $\{f_{\hat{v}} | \hat{v} \in \mathcal{N}_v\}$

Output: New embedding f_v^* for entity v

$h_{\hat{v}} \leftarrow AGG.(f_{\hat{v}}, \hat{v} \in \mathcal{N}_v)$;

$f_v^{la} \leftarrow ReLU(W_1 h_{\hat{v}} + b_1)$;

$f_v \leftarrow ReLU(W_2 \cdot CONCAT(f_v^{in}, f_v^{la}) + b_2)$;

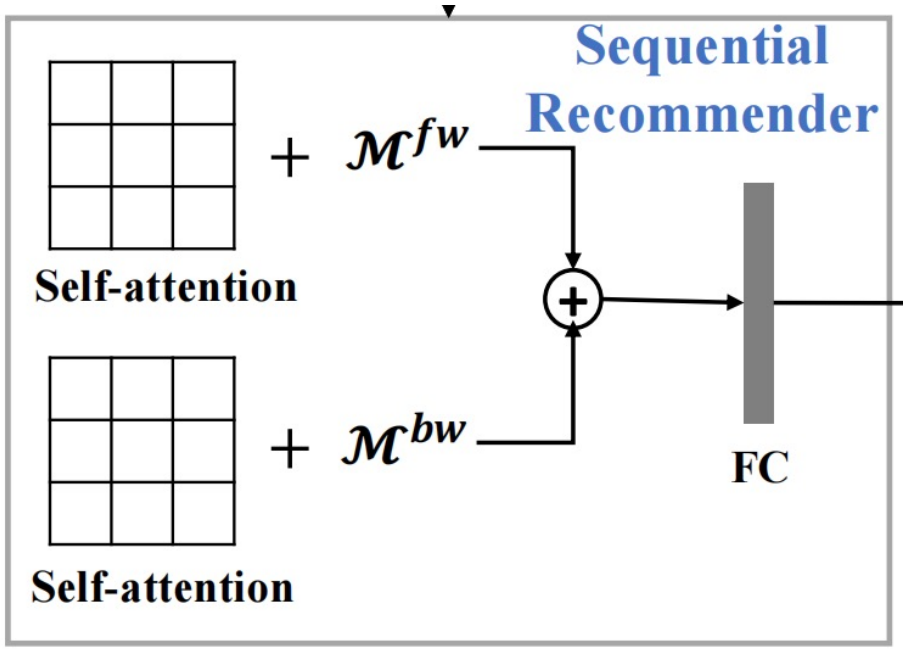
$f_v^* \leftarrow f_v / \|f_v\|_2$

- The Diffusion Representer, which works on the user-item interaction graph, is to learn the users' and items' high-order interactive representation.

- ***inherent feature*** vector of the entity itself, which can incorporate one-hot ID, attributes, context information, and so on.
- ***latent feature*** vector obtained through information diffusion, which contains the structural information of the interaction graph and the information supplement of neighbor nodes.

Huang X, Sang J, Xu C. Learning to Learn a Sequential Recommender[J]. IEEE Transactions on Information Systems (TOIS), 2021, Accepted. Arxiv Link: https://arxiv.org/abs/2110.09083.

metaCSR: Sequential Recommender



$$M_{m,n}^{fw} = \begin{cases} -|d_{m,n}|, & m < n \\ -\infty, & otherwise \end{cases}$$

$$M_{m,n}^{bw} = \begin{cases} -|d_{m,n}|, & m > n \\ -\infty, & otherwise \end{cases}$$

Xiaowen Huang, Shengsheng Qian, Quan Fang, Jitao Sang, and Changsheng Xu. CSAN: Contextual Self-Attention Network for User Sequential Recommendation. In Proceedings of the 26th ACM international conference on Multimedia (MM '18)

Huang X, Sang J, Xu C. Learning to Learn a Sequential Recommender[J]. IEEE Transactions on Information Systems (TOIS), 2021, Accepted. Arxiv Link: https://arxiv.org/abs/2110.09083.

metaCSR: Meta Learner

In the ***meta-train* phase:**

$$\mathcal{L} = \sum_{u} \sum_{j} \sum_{j^- \in \mathcal{J}^-} -log[p_{u,j} - p_{u,j^-}]$$

Ranking the ground-truth item j higher than all other items.

In the ***inner loop optimization*** procedure:

$$\min_{\theta_2} \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_1,\theta_2})$$

$$\theta_2^{'} = \theta_2 - \alpha \nabla_{\theta_2} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_1,\theta_2})$$

Updating the task-specific model, i.e., the Sequential Recommender, by one or more gradient descent updates using support set.

In the ***outer loop optimization*** procedure

$$\min_{\theta_!,\theta_2} \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_1,\theta_2^{'}}) = \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_1,\theta_2 - \alpha \nabla_{\theta_2} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_1,\theta_2})})$$

Updating all parameters by optimizing the performance $f_{\theta_1,\theta_2^{'}}$ using query set.

$$\theta_1 \leftarrow \theta_1 - \beta \nabla_{\theta_1} \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_1,\theta_2^{'}})$$

$$\theta_2 \leftarrow \theta_2 - \beta \nabla_{\theta_2} \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_1,\theta_2^{'}})$$

Huang X, Sang J, Xu C. Learning to Learn a Sequential Recommender[J]. IEEE Transactions on Information Systems (TOIS), 2021, Accepted. Arxiv Link: https://arxiv.org/abs/2110.09083.

metaCSR: Meta Learner

In the ***meta-test phase*** **:**

a few samples of new users are applied to fine-tune the trained model, and then the new model is applied to perform the final recommendations.

Algorithm 2: metaCSR

Input: $p(\mathcal{T})$: distribution over tasks;
α, β: step size hyperparameters;
θ_1: parameters of Diffusion Representer;
θ_2: parameters of Sequential Recommender;

Initialize θ_1, θ_2 randomly;
while *not converged* **do**
  Sample batch of tasks $\mathcal{T}_i \sim p(\mathcal{T})$
  for *all* $\mathcal{T}_i$ **do**
    Sample K_1 sequences as support set D_s from $\mathcal{T}_i$
    Evaluate $\nabla_{\theta_2} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_1,\theta_2})$ using D_s
    Inner loop optimization:
    Compute adapted parameters with gradient descent:
    $\theta_2' = \theta_2 - \alpha \nabla_{\theta_2} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_1,\theta_2})$
  end
  Outer loop optimization:
  Sample K_2 sequences as query set D_Q from $\mathcal{T}_i$
  Update θ_1, θ_2 using D_Q:
  $\theta_1 \leftarrow \theta_1 - \beta \nabla_{\theta_1} \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_1,\theta_2'})$
  $\theta_2 \leftarrow \theta_2 - \beta \nabla_{\theta_2} \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_1,\theta_2'})$
end

Huang X, Sang J, Xu C. Learning to Learn a Sequential Recommender[J]. IEEE Transactions on Information Systems (TOIS), 2021, Accepted. Arxiv Link: https://arxiv.org/abs/2110.09083.

Experimental Results

Table 3. The next-one recommendation performance of all the methods across the evaluation metrics AUC and MAP in **COLD-START scenario**. The best performance is boldfaced; the highest score in baseline is labeled with '*'; the percentage in parentheses (+/-%) represents the relative improvements that metaCSR achieve w.r.t the best baseline.

Datasets	Methods	Evaluation Metrics	
		AUC	MAP
MovieLens-1M	BPR	0.7355	0.2031
	LSTM	0.7621	0.2119
	CSAN	0.7693	0.2563*
	MeLU	0.7391	0.1964
	MetaCS-L	0.7029	0.2100
	MetaCS-DNN	0.7035	0.2009
	SML	0.7621	0.1501
	$\text{MetaCF}_{\text{NGCF}}$	0.7995*	0.2506
	metaCSR	**0.8623** (+7.85%)	**0.2987** (+16.54%)
Last.fm-1week	BPR	0.8333	0.2639
	LSTM	0.8889	0.3903
	CSAN	0.9242*	0.4829
	MeLU	0.8586	0.2900
	MetaCS-L	0.8485	0.3122
	MetaCS-DNN	0.8788	0.5051*
	SML	0.9053	0.4940
	$\text{MetaCF}_{\text{NGCF}}$	0.9242*	0.5005
	metaCSR	**0.9596** (+3.83%)	**0.5394** (+6.79%)
Amazon-Video	BPR	0.7217	0.1859
	LSTM	0.7831	0.2494
	CSAN	0.7791	0.2509
	MeLU	0.7580	0.2461
	MetaCS-L	0.7207	0.2419
	MetaCS-DNN	0.7213	0.2566*
	SML	0.7997	0.2444
	$\text{MetaCF}_{\text{NGCF}}$	**0.8041***	0.2530
	metaCSR	0.8011 (-0.37%)	**0.3326** (+29.62%)

Table 4. The next-one recommendation performance of all the methods across the evaluation metrics AUC and MAP in **WARM-START scenario**. The best performance is boldfaced; the highest score in baseline is labeled with '*'; the percentage in parentheses (+/-%) represents the relative improvements that metaCSR achieve w.r.t the best baseline.

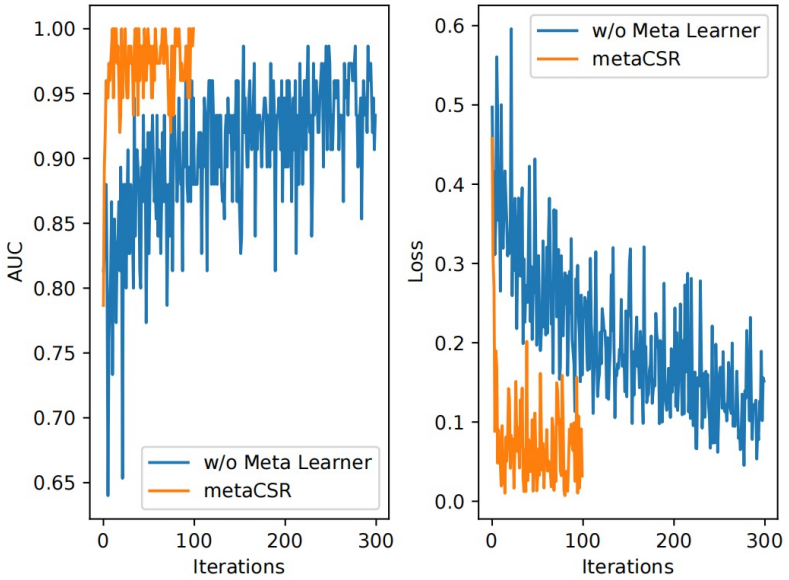
Datasets	Methods	Evaluation Metrics	
		AUC	MAP
MovieLens-1M	BPR	0.8054	0.2079
	LSTM	0.8272	0.2253
	CSAN	0.8414*	0.2596*
	MeLU	0.7722	0.1885
	MetaCS-L	0.7446	0.2160
	MetaCS-DNN	0.7371	0.2171
	SML	0.7935	0.1609
	$\text{MetaCF}_{\text{NGCF}}$	0.8347	0.2134
	metaCSR	**0.8589**(+2.08%)	**0.2742**(+5.62%)
Last.fm-1week	BPR	0.9318	0.5075
	LSTM	0.9545	0.5455
	CSAN	0.9798	0.5860
	MeLU	0.9848*	0.7156
	MetaCS-L	0.9545	0.7880
	MetaCS-DNN	0.9545	0.8404*
	SML	0.9583	0.8009
	$\text{MetaCF}_{\text{NGCF}}$	0.9697	0.8230
	metaCSR	**0.9886**(+0.39%)	**0.9097**(+7.37%)
Amazon-Video	BPR	0.8107	0.3532
	LSTM	**0.8432***	0.3900
	CSAN	0.8375	**0.4483***
	MeLU	0.7522	0.2646
	MetaCS-L	0.8301	0.2748
	MetaCS-DNN	0.8285	0.2728
	SML	0.8363	0.2824
	$\text{MetaCF}_{\text{NGCF}}$	0.8427	0.2831
	metaCSR	0.8419(-0.15%)	0.3813(-14.95%)

Huang X, Sang J, Xu C. Learning to Learn a Sequential Recommender[J]. IEEE Transactions on Information Systems (TOIS), 2021, Accepted. Arxiv Link: https://arxiv.org/abs/2110.09083.

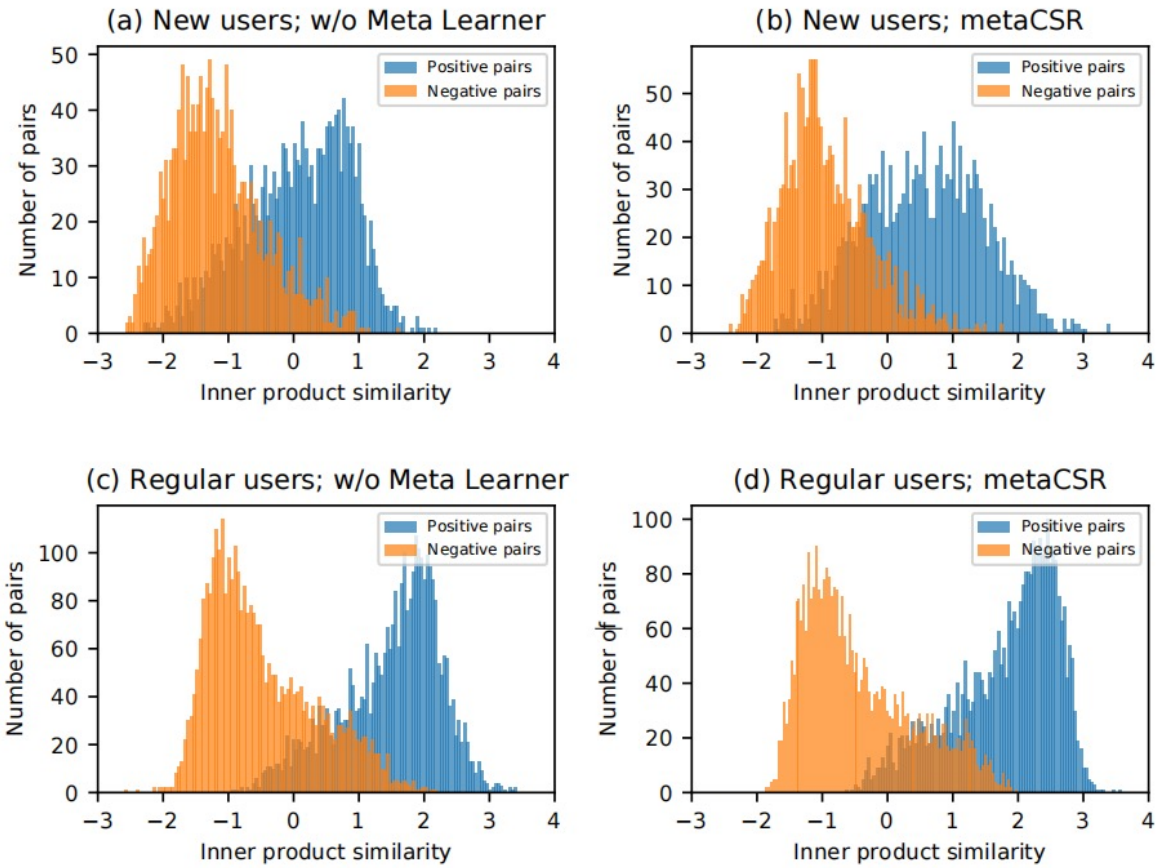
Experimental Results

Ablation studies on MovieLens-1M dataset.

Index	Diffusion Representer	Sequential Recommender	Meta Learner	AUC	Improve%
1	×	✓	✓	0.8285	-3.92%
2	✓	×	✓	0.8297	-3.78%
3	✓	✓	×	0.8056	-6.58%
4	✓	✓	✓	**0.8623**	-



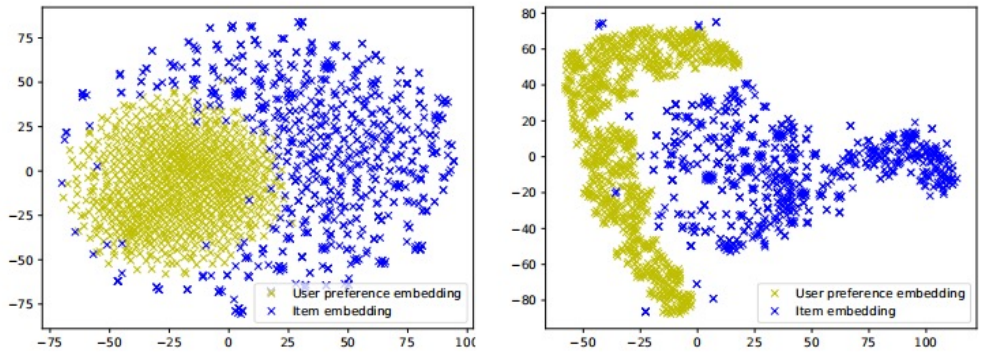
AUC and Loss change with the number of iterations increasing. It is validated on MovieLens-1M dataset.



Distributions of inner product similarity of positive pairs and negative pairs in different scenarios. It is validated on MovieLens-1M dataset.

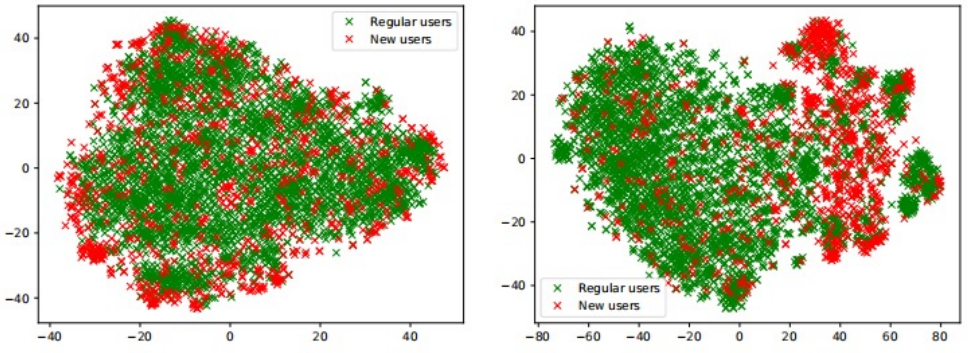
Huang X, Sang J, Xu C. Learning to Learn a Sequential Recommender[J]. IEEE Transactions on Information Systems (TOIS), 2021, Accepted. Arxiv Link: https://arxiv.org/abs/2110.09083.

Experimental Results



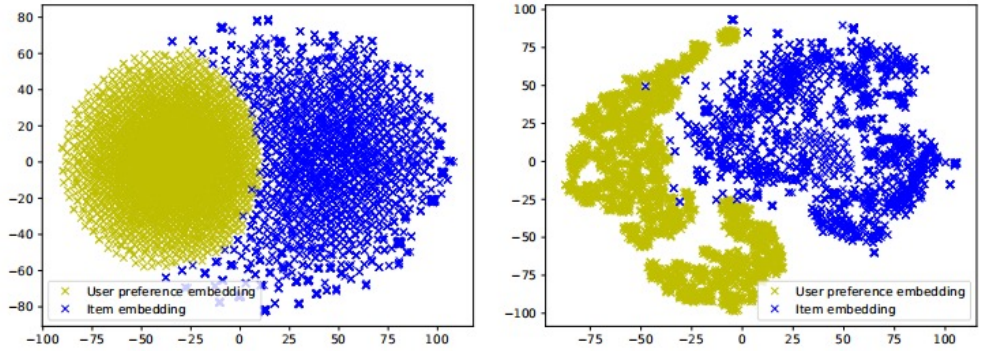
(a) metaCSR (b) w/o Meta Learner

Fig. 7. Visualization of **NEW** users' preference embedding and items embedding.



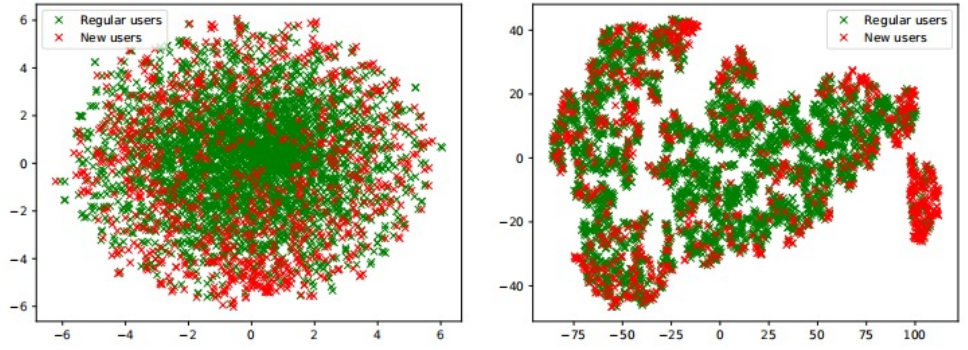
(a) metaCSR (b) w/o Meta Learner

Fig. 10. Visualization of new/regular users' preference embedding **WITHOUT fine-tuning**.



(a) metaCSR (b) w/o Meta Learner

Fig. 8. Visualization of **REGULAR** users' preference embedding and items embedding.



(a) metaCSR (b) w/o Meta Learner

Fig. 11. Visualization of new/regular users' preference embedding **WITH fine-tuning**.

Huang X, Sang J, Xu C. Learning to Learn a Sequential Recommender[J]. IEEE Transactions on Information Systems (TOIS), 2021, Accepted. Arxiv Link: https://arxiv.org/abs/2110.09083.

Robustness

Robustness Issues

Internal affair

- Data: Noise, unbalanced data, missing data
- Model: Parameter sensitivity

External affair

- Attack and Defense
  - Data poisoning/shilling attacks: promote/demote a set of items

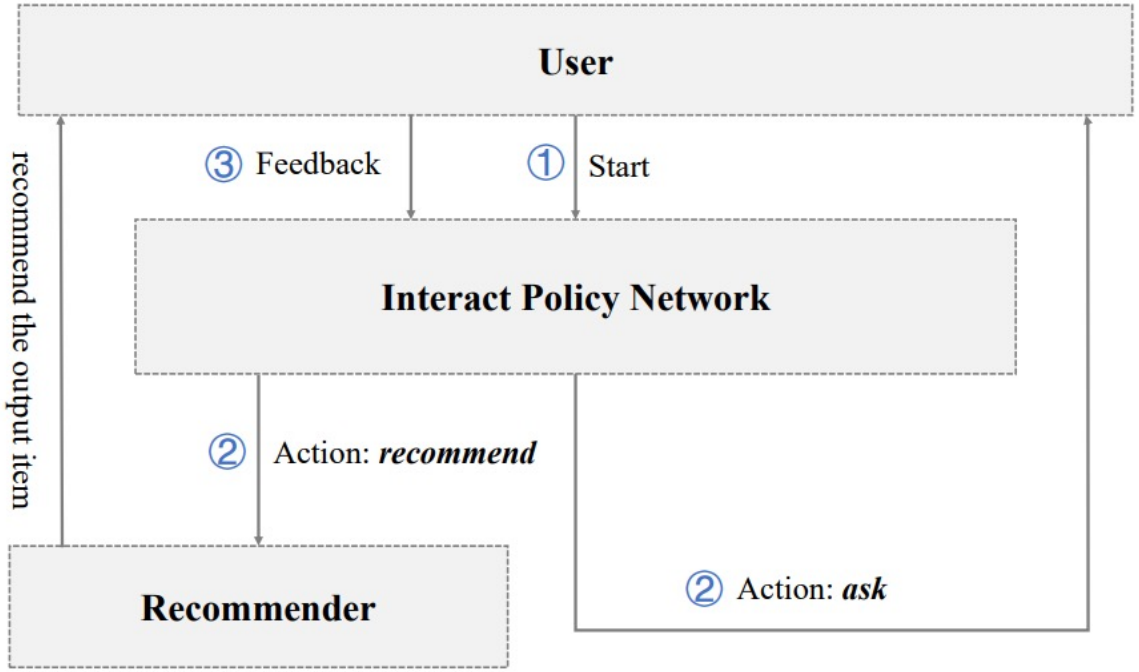
Active user modeling

→ **Conversational recommendation**

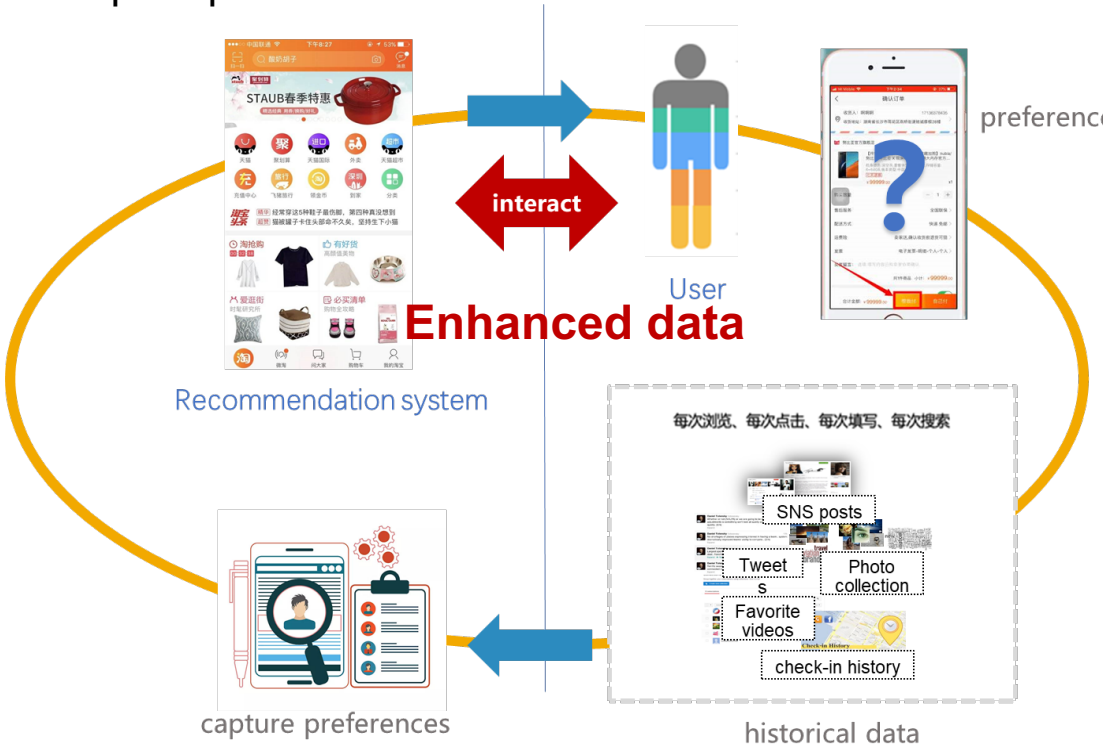
@Min Zhang

KGenSam

- **Task**: Efficient conversational recommendation.
- **Motivation:** Since the environmental information modeling is insufficient, our work introduces **KG** into CRS as the environment, enhances the knowledge of environment, and focus on solving two specific problems by sampling methods:
  - (1) What **attributes** should CRS **ask** to help the recommender fit the current user's preferences **efficiently**?
  - (2) How can CRS **update the recommender precisely** to fit the implicit preference information in user feedback?



The basic framework and workflow of multi-round CRS

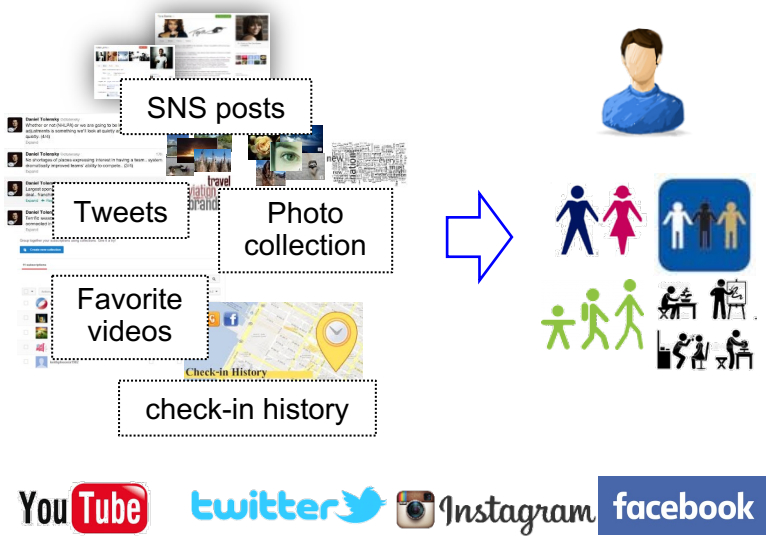


Unreliable estimation

Noise, unbalanced data

Mengyuan Zhao, Xiaowen Huang, Lixi Zhu, Jitao Sang, Jian Yu. Knowledge Graph-enhanced Sampling for Conversational Recommendation System. TKDE under review.
arXiv link: http://arxiv.org/abs/2110.06637

KGenSam



Noisy data

Irrelevant behaviors and context sensitive behaviors

Active Learning → **Fuzzy Samples with High Confidence**

Unbalanced data

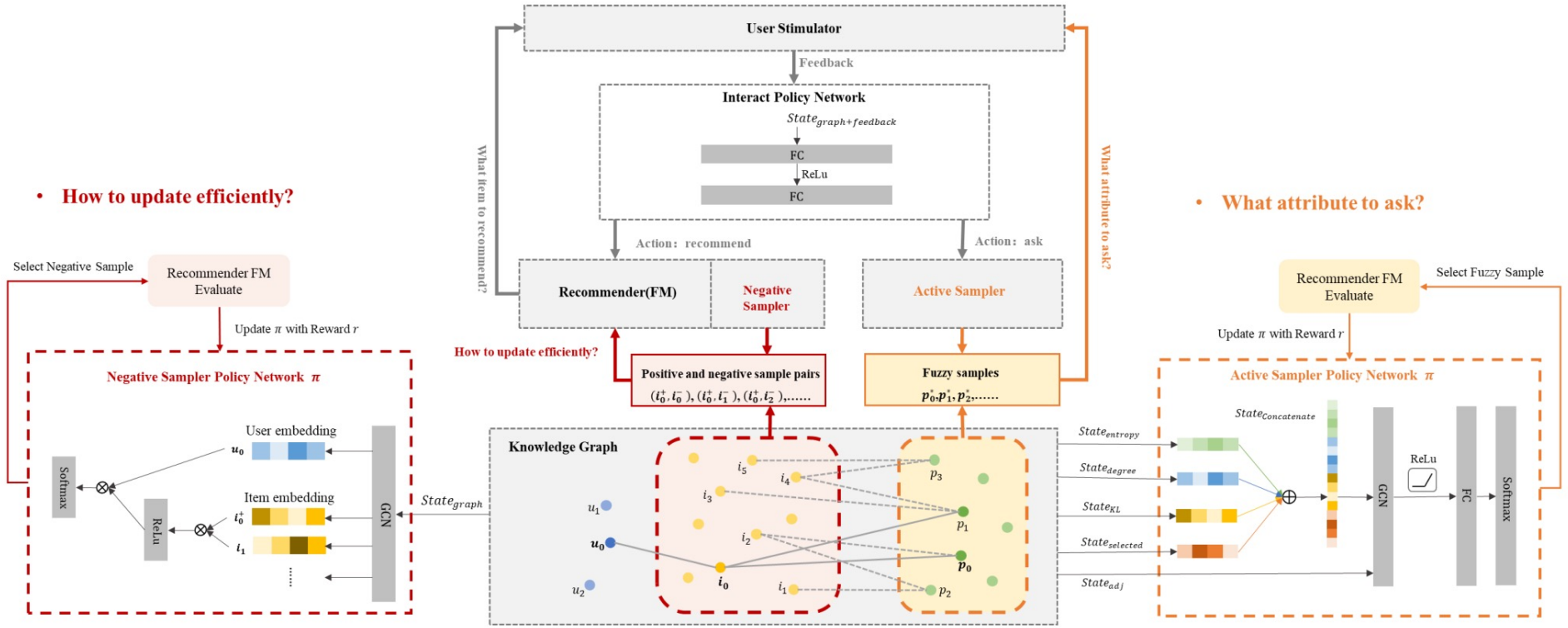
Only positive samples, no explicit negative samples

Negative Sampling → **Hard-negative Samples with Large Gradient**

Mengyuan Zhao, Xiaowen Huang, Lixi Zhu, Jitao Sang, Jian Yu. Knowledge Graph-enhanced Sampling for Conversational Recommendation System. TKDE under review.

arXiv link: http://arxiv.org/abs/2110.06637

KGenSam: Model



The key design is two sampler modules:

- **Active Sampler** outputs **fuzzy item attribute samples with high uncertainty** as the user preference information to be asked if CRS takes the ask action.
- **Negative Sampler** outputs **high-quality negative item samples** and constructs positive and negative sample pairs to the effective update of the recommender at each turn of conversation.

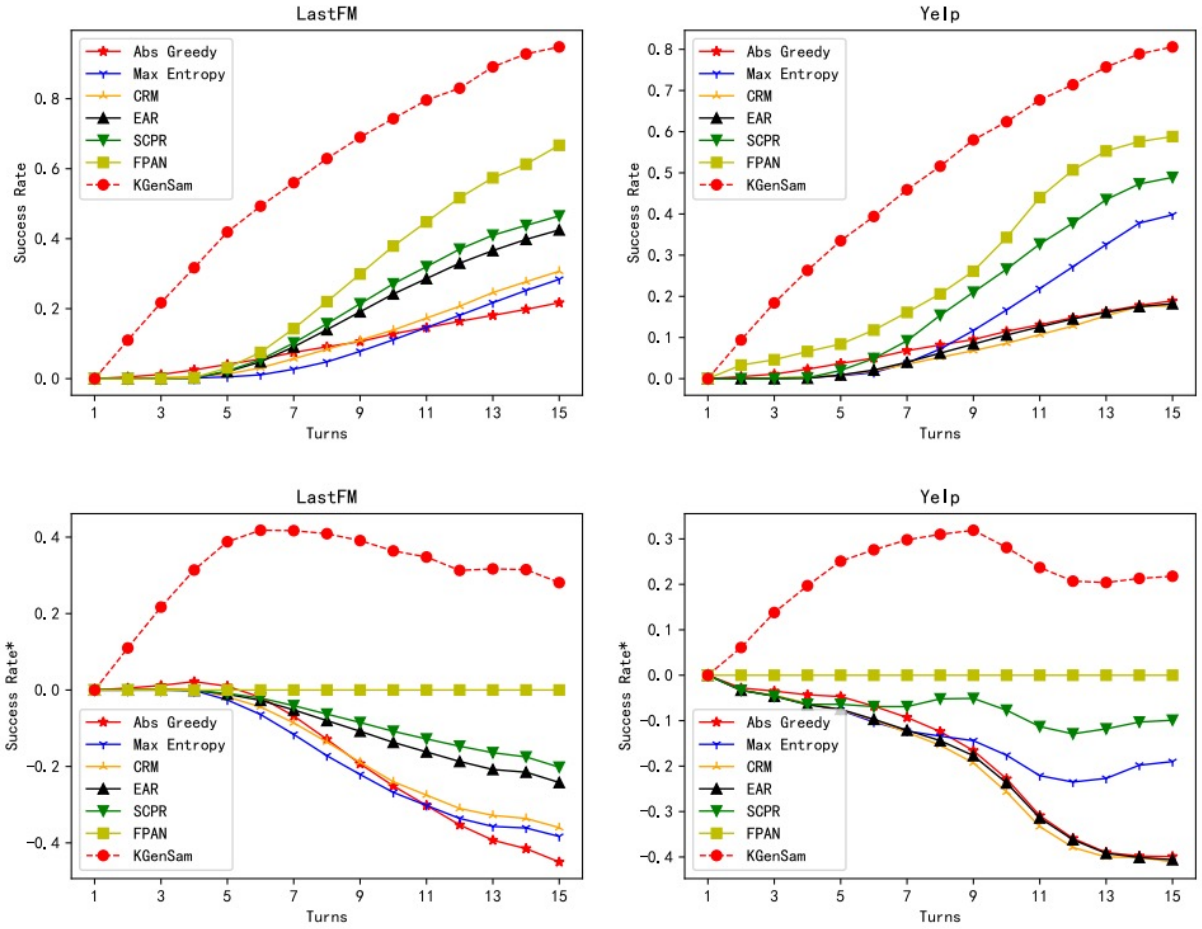
Mengyuan Zhao, Xiaowen Huang, Lixi Zhu, Jitao Sang, Jian Yu. Knowledge Graph-enhanced Sampling for Conversational Recommendation System. TKDE under review.

arXiv link: http://arxiv.org/abs/2110.06637

Experimental Results

TABLE 2
Success Rate@15 and Average Turns

	LastFM		Yelp	
	SR@15	**AT**	**SR@15**	**AT**
Abs Greedy	0.222	13.48	0.189	13.43
Max Entropy	0.283	13.91	0.398	13.42
CRM	0.325	13.75	0.177	13.69
EAR	0.429	12.88	0.182	13.63
SCPR	0.465	12.86	0.489	12.62
FPAN	0.667	10.14	0.588	12.65
KGenSam	**0.948**	**6.43**	**0.810**	**7.72**
improve	42.0%↑	36.6%↑	37.8%↑	38.8%↑



Efficient CRS：

- **Fewer interaction rounds**
- **Higher recommendation success rate**

Mengyuan Zhao, Xiaowen Huang, Lixi Zhu, Jitao Sang, Jian Yu. Knowledge Graph-enhanced Sampling for Conversational Recommendation System. TKDE under review.

arXiv link: http://arxiv.org/abs/2110.06637

Robustness

Robustness Issues

Internal affair

- Data: Noise, unbalanced data, missing data
- Model: Parameter sensitivity

External affair

- Attack and Defense
  - Data poisoning/shilling attacks: promote/demote a set of items

- To increase the robustness of recommendation models with the existence of shilling attacks
- To detect and block fraudsters from the data.

@Min Zhang

Attacks in The Real World

Taiwan fines Samsung $340,000 for bashing HTC

The country's Fair Trade Commission determined Samsung misled consumers by hiring students to post fake comments about Samsung and HTC phones.

Shara Tibken Oct. 24, 2013 11:01 a.m. PT

Samsung's Galaxy S4 rivals the HTC One.

- In 2015, Samsung was alleged by Taiwan's Fair Trade Commission to have hired students to post negative comments about HTC phones.

Customer product reviews policies

Customer reviews are an integral part of the customer shopping experience on Amazon. Customers use these reviews to learn more about the product, assess whether it fits their needs, and make an informed purchase decision. Customer reviews also help sellers understand the customers' sentiment about their products, what features or aspects of the product customers like, and what areas need improvements. Reviews also provide sellers with ideas on how to improve their products. In order for customer reviews to continue to provide these benefits to customers and sellers, they have to remain a true and authentic reflection of customers' experiences with the products.

Amazon's Community Guidelines have specific policies that are meant to protect the authenticity of Customer Reviews, and we ask you to comply with these policies and report any violations you might notice.

We strongly urge you to thoroughly review Amazon Customer Reviews policies and immediately correct any violating actions. It is important that you educate your business partners, employees, and any third-party partners you work with about these policies as well. Any infractions by your business partners, employees, or third party agencies will result in enforcement actions, even if it happened without your knowledge or consent.

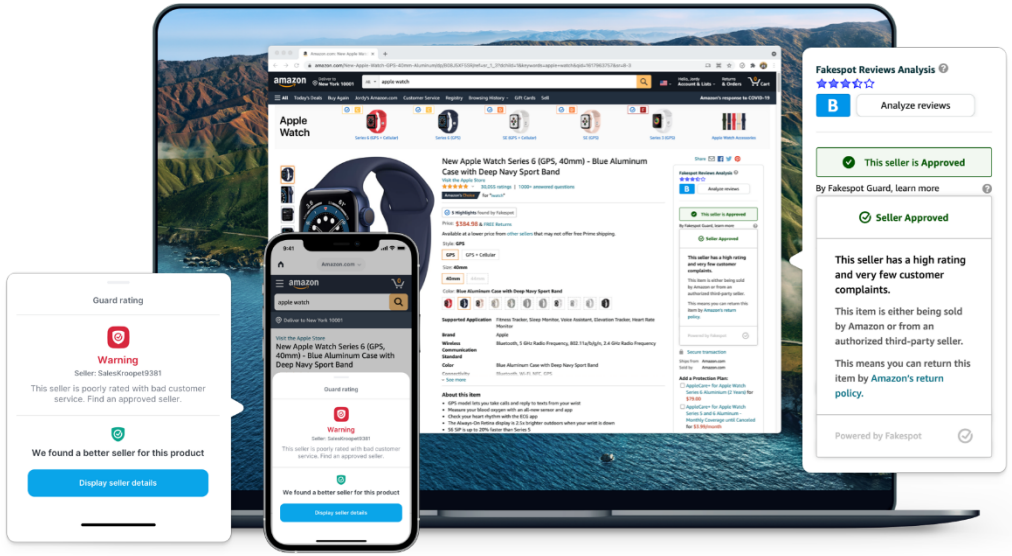
Violations to Customer Reviews policies include, but are not limited to, these actions:

- In order for customer reviews to continue to provide these benefits to customers and sellers, they have to remain an authentic reflection of customers' experiences with the products.

https://www.cnet.com/tech/mobile/taiwan-fines-samsung-340000-for-bashing-htc/
https://sellercentral.amazon.com/gp/help/external/YRKB5RU3FS5TURN?language=en_US

Attacks in The Real World

- A RAVPower charging brick came with an offer for a gift card in exchange for a review[1]



- About **42%** of 720 million Amazon reviews assessed by the monitoring service **Fakespot** from March through September were unreliable, up from about **36%** for the same period last year. The rise in fake reviews corresponded with the stampede online of millions of virus-avoiding shoppers.

Shilling attack

Need to be defended

[1]https://www.wsj.com/articles/fake-reviews-and-inflated-ratings-are-still-a-problem-for-amazon-11623587313
[2]https://www.chicagotribune.com/business/ct-biz-amazon-fake-reviews-unreliable-20201020-lfbjdq25azfdpa3iz6hn6zvtwq-story.html

Attack Types

Shilling attack:

- Malicious users and/or rival companies might try to insert fake profiles into user-item matrices in order to affect the predicted ratings and/or diminish the performance of the system on behalf of their goals.
- Some attacks might intend to increase the popularity of some targeted items (referred to as the push attack) while some others might aim to decrease the popularity of some targeted items (referred to as the nuke attack) .

Table 2 Attack types according to intent and required knowledge dimensions

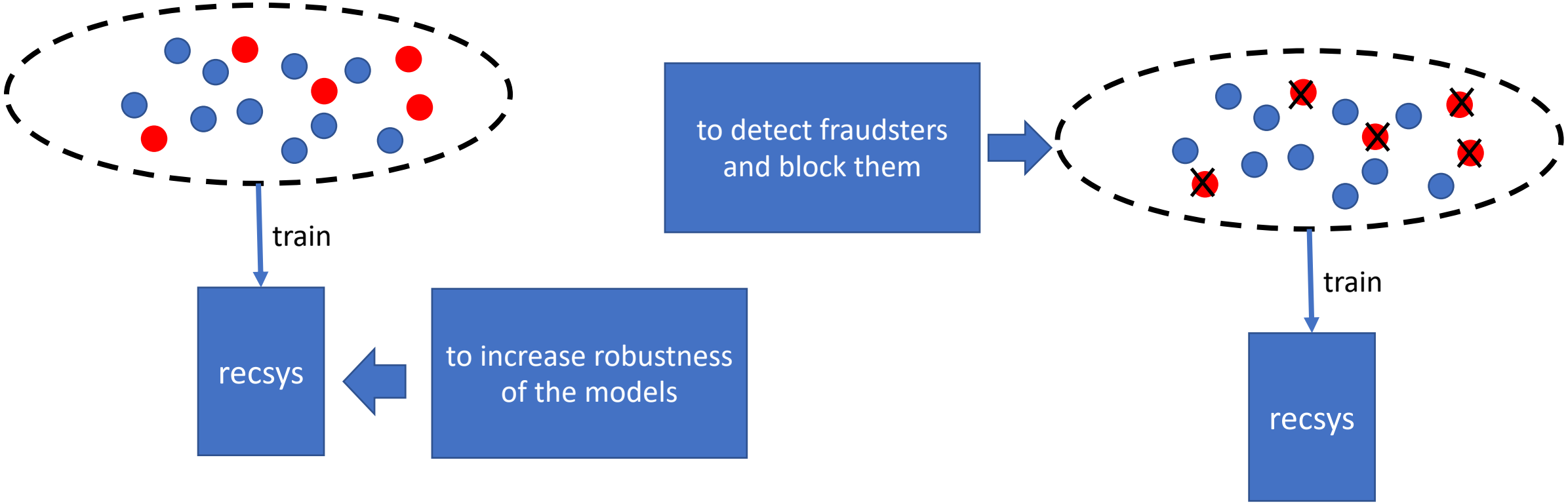
Attack type	Intent		Required knowledge		
	Push	Nuke	Low	High	Informed
Random (RandomBot)	✓	✓	✓		
Average (AverageBot)	✓	✓		✓	
Probe	✓	✓			✓
Bandwagon (Popular)	✓		✓		
Segment	✓		✓		
Reverse Bandwagon		✓	✓		
Love/hate		✓	✓		
Hybrid	✓	✓	✓		
Consistency (favorite item)	✓	✓		✓	
Perfect knowledge	✓	✓		✓	

Shilling attacks against recommender systems: a comprehensive survey

Attack and Defense

How to defense shilling attack? Two pathways:

- To increase the robustness of recommendation models with the existence of shilling attacks
- To detect and block fraudsters from the data



Zhang S, Yin H, Chen T, et al. Gcn-based user representation learning for unifying robust recommendation and fraudster detection[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2020: 689-698.

Increase the Robustness of the model: SGL

- **Task**: Learning high-quality user and item representations from interaction data.
- **Motivation**: three limitations of GCN-based recommender:
  - Sparse Supervision Signal.
  - Skewed Data Distribution.
  - Noises in Interactions.

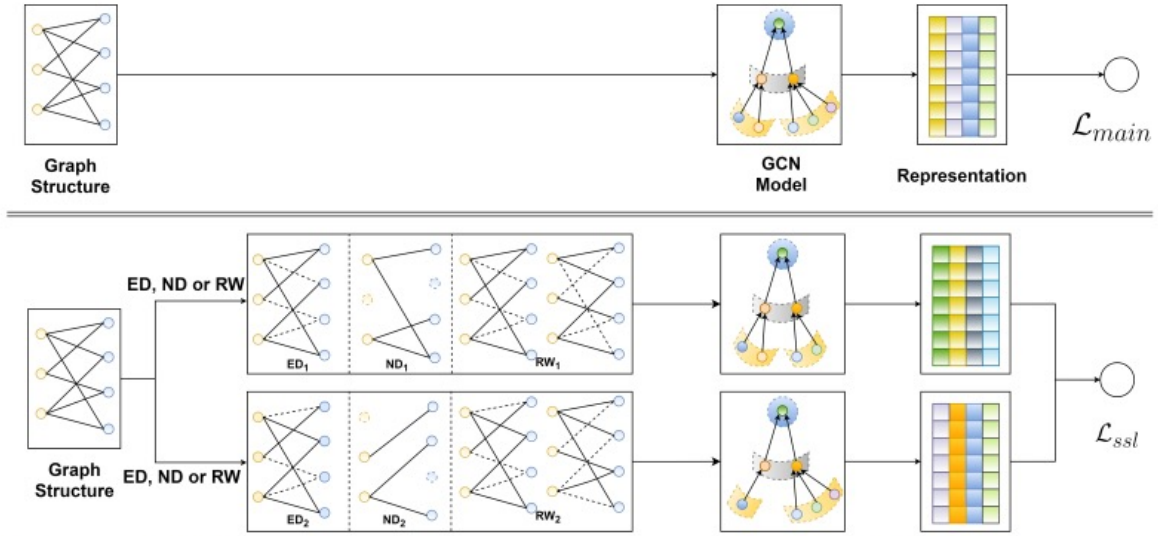


Figure 1: The overall system framework of SGL. The upper layer illustrates the working flow of the main supervised learning task while the bottom layer shows the working flows of SSL task with augmentation on graph structure.

Three augmentation operators:

- **Node Dropout (ND)**

$$s_1(\mathcal{G}) = (\mathbf{M}' \odot \mathcal{V}, \mathcal{E}), \quad s_2(\mathcal{G}) = (\mathbf{M}'' \odot \mathcal{V}, \mathcal{E}),$$

- **Edge Dropout (ED)**

$$s_1(\mathcal{G}) = (\mathcal{V}, \mathbf{M}_1 \odot \mathcal{E}), \quad s_2(\mathcal{G}) = (\mathcal{V}, \mathbf{M}_2 \odot \mathcal{E}),$$

- **Random Walk (RW)**

$$s_1(\mathcal{G}) = (\mathcal{V}, \mathbf{M}_1^{(l)} \odot \mathcal{E}), \quad s_2(\mathcal{G}) = (\mathcal{V}, \mathbf{M}_2^{(l)} \odot \mathcal{E}),$$

$$\mathcal{L}_{ssl}^{user} = \sum_{u \in \mathcal{U}} -\log \frac{\exp(s(\mathbf{z}_u', \mathbf{z}_u'')/\tau)}{\sum_{v \in \mathcal{U}} \exp(s(\mathbf{z}_u', \mathbf{z}_v'')/\tau)}, \qquad \mathcal{L}_{ssl} = \mathcal{L}_{ssl}^{user} + \mathcal{L}_{ssl}^{item}.$$

$$\mathcal{L}_{main} = \sum_{(u,i,j) \in O} -\log \sigma(\hat{y}_{ui} - \hat{y}_{uj}),$$

Multi-task Training

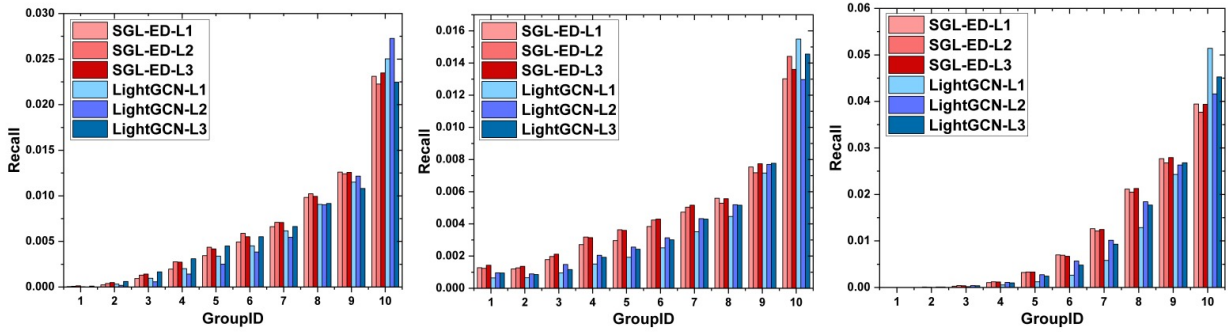
$$\mathcal{L} = \mathcal{L}_{main} + \lambda_1 \mathcal{L}_{ssl} + \lambda_2 \|\Theta\|_2^2,$$

Wu J, Wang X, Feng F, et al. Self-supervised graph learning for recommendation[C]//Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2021: 726-735.

Results

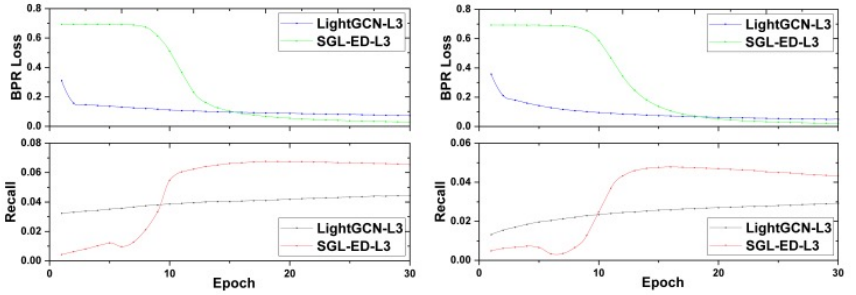
Table 4: Overall Performance Comparison.

Dataset	Yelp2018		Amazon-Book		Alibaba-iFashion	
Method	Recall	NDCG	Recall	NDCG	Recall	NDCG
NGCF	0.0579	0.0477	0.0344	0.0263	0.1043	0.0486
LightGCN	0.0639	0.0525	0.0411	0.0315	0.1078	0.0507
Mult-VAE	0.0584	0.0450	0.0407	0.0315	0.1041	0.0497
DNN+SSL	0.0483	0.0382	0.0438	0.0337	0.0712	0.0325
SGL-ED	**0.0675**	**0.0555**	**0.0478**	**0.0379**	**0.1126**	**0.0538**
%Improv.	5.63%	5.71%	9.13%	12.46%	4.45%	6.11%
p-value	5.92e-8	1.89e-8	5.07e-10	3.63e-10	3.34e-8	4.68e-10



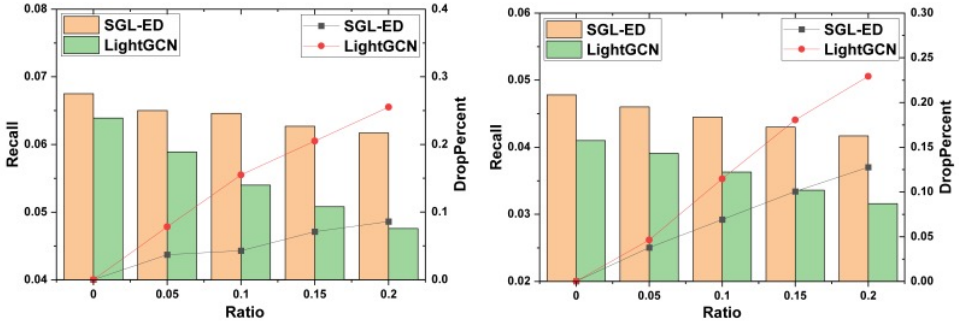
(a) Yelp2018 (b) Amazon-Book (c) Alibaba-iFashion

Figure 4: Performance comparison over different item groups between SGL-ED and LightGCN. The suffix in the legend indicates the number of GCN layers.



(a) Yelp2018-L3 (b) Amazon-Book-L3

Figure 5: Training Curves of SGL-ED and LightGCN on three datasets. The suffix in the legend denotes the layer numbers.



(a) Yelp2018 (b) Amazon-Book

Figure 6: Model performance *w.r.t.* noise ratio. The bar represents Recall, while the line represents the percentage of performance degradation.

Detect and Block Fraudsters: GraphRfi

- **Task**: Defense shilling attack
- **Motivation**: The review rating data for a recommender system typically comes from open platforms, which may attract a group of malicious users to deliberately insert fake feedback in an attempt to bias the recommender system to their favour.
- **Method**:
  - GCN: Rating prediction
  - Neural Random Forests: Fraudster detection

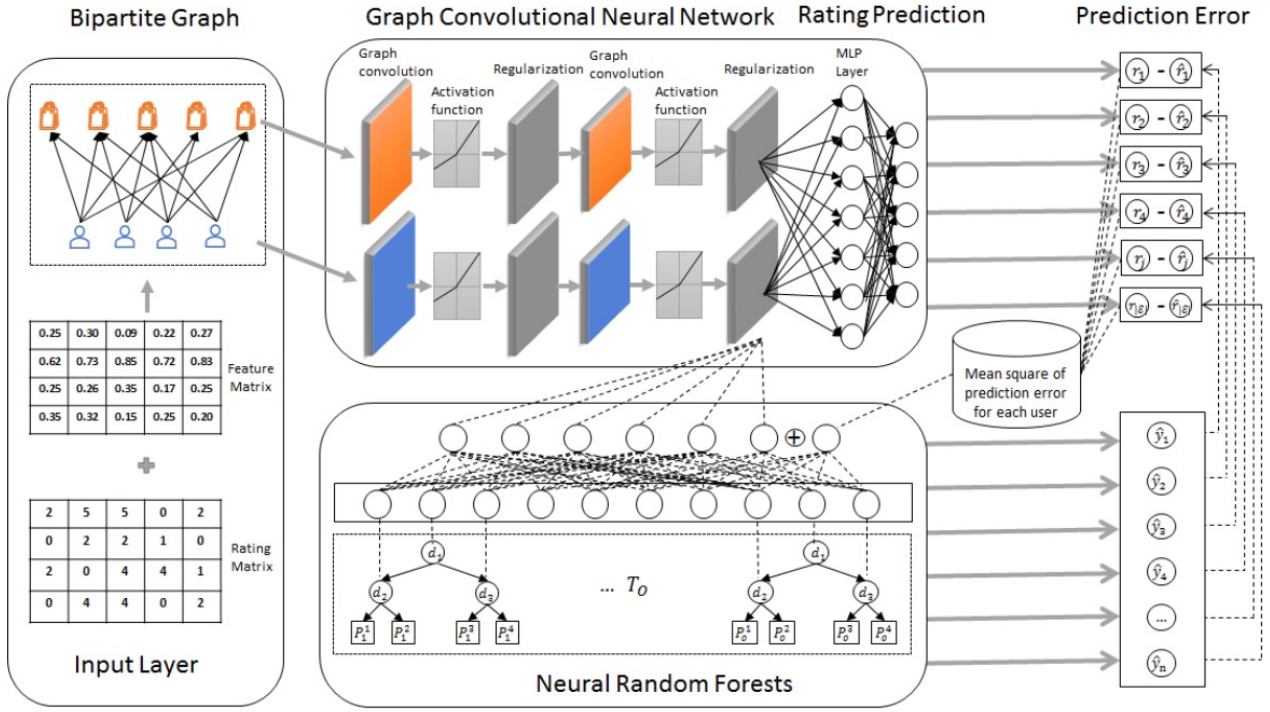


Figure 1: The overview of GraphRfi

$$\mathcal{L}_{rating} = \frac{1}{|\mathcal{E}|} \sum_{\forall u,v \in \mathcal{E}} \mathbb{P}_T[y = 0 \mid \mathbf{z}_u^*, \Theta, \pi] \cdot (r'_{uv} - r_{uv})^2$$

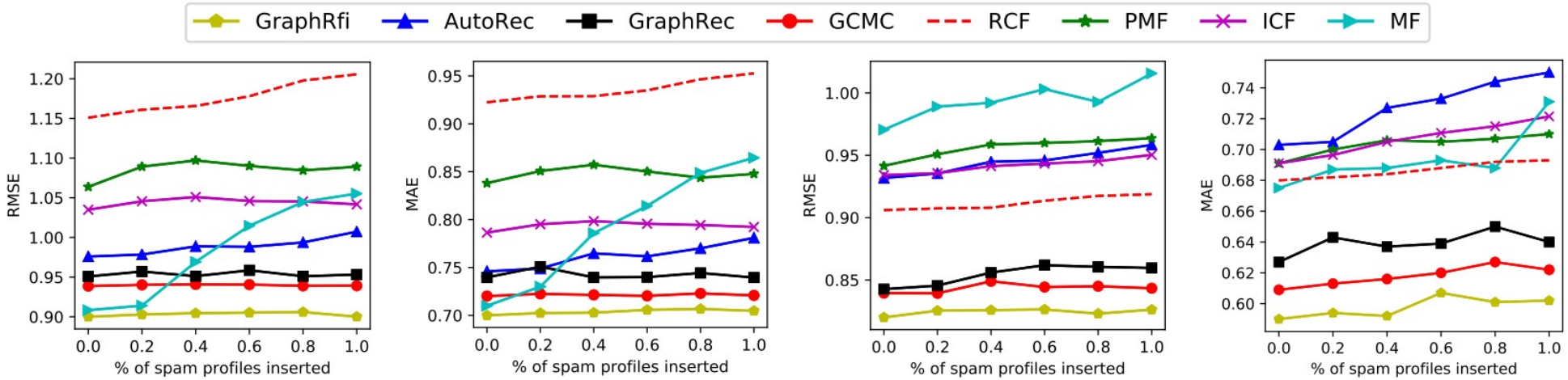
$$\mathcal{L}_{fraudster} = \frac{1}{|\mathcal{U}|} \sum_{\forall u \in \mathcal{U}, y_u \in \mathcal{Y}} -\log \mathbb{P}_T[y = y_u \mid \mathbf{z}_u^*, \Theta, \pi]$$

Multi-task Training

$$\mathcal{L} = \mathcal{L}_{rating} + \lambda \mathcal{L}_{fraudster}$$

Zhang S, Yin H, Chen T, et al. Gcn-based user representation learning for unifying robust recommendation and fraudster detection[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2020: 689-698.

Robustness



(a) RMSE on Yelp (b) MAE on Yelp (c) RMSE on Movies & TV (d) MAE on Movies & TV

Figure 2: Rating prediction results on Yelp and Movies & TV. Spam profiles refer to fraudsters in datasets.

Dataset	Method	Mixed Attack			Hate Attack			Average Attack			Random Attack		
		Precision	Recall	F1 Score	Precision	Recall	F1 Score	Precision	Recall	F1 Score	Precision	Recall	F1 Score
Yelp	Rev2	0.783	0.680	0.730	0.681	0.941	0.790	0.874	0.869	0.871	0.908	0.935	0.921
	DegreeSAD	0.923	0.918	0.921	0.926	0.919	0.922	0.922	0.917	0.919	0.922	0.920	0.921
	FAP	0.688	0.013	0.026	0.796	0.015	0.030	0.742	0.014	0.028	0.711	0.014	0.027
	OFD	0.941	0.991	0.966	0.935	**0.998**	0.966	0.983	0.982	0.982	0.979	**0.993**	0.986
	GraphRFI	**0.989**	**0.992**	**0.990**	**0.979**	0.995	**0.987**	**0.993**	**0.991**	**0.992**	**0.987**	0.992	**0.990**
Movie & TV	Rev2	0.554	0.565	0.560	0.820	0.876	0.847	0.779	0.941	0.852	0.571	0.671	0.617
	DegreeSAD	0.636	0.631	0.633	0.627	0.619	0.623	0.624	0.626	0.625	0.626	0.607	0.615
	FAP	0.527	0.249	0.338	0.515	0.246	0.333	0.535	0.258	0.349	0.557	0.266	0.360
	OFD	0.940	0.981	0.960	0.945	0.983	0.964	0.977	0.980	0.975	0.954	0.985	0.969
	GraphRFI	**0.967**	**0.986**	**0.976**	**0.965**	**0.989**	**0.977**	**0.978**	**0.993**	**0.985**	**0.963**	**0.996**	**0.979**

Table 3: Fraudster Detection Performance on Yelp and Movies & TV.

Evaluations for Robustness

- **Case study** [1]
  - Present a snapshot of the interaction sequence for a sampled user and use different models to recommend. Finally, compare Intuitively the results of the two methods.
- **Add noise** [2]
  - Contaminate the training set by adding a certain proportion of adversarial examples (i.e., 5%, 10%, 15%, 20% negative user-item interactions), while keeping the testing set unchanged.
- **Attack** [3]
  - Rank shift, which is defined as the difference between the specific item's rank before and after the attack.
- **Detection acc. [**4][5]
  - Rank the users based on how fraudulent they are.
- **Sparsity** [6]
  - Train a model with only partial training data (like 25%, 50%, 75%, and 100%) and keep the test data unchanged.

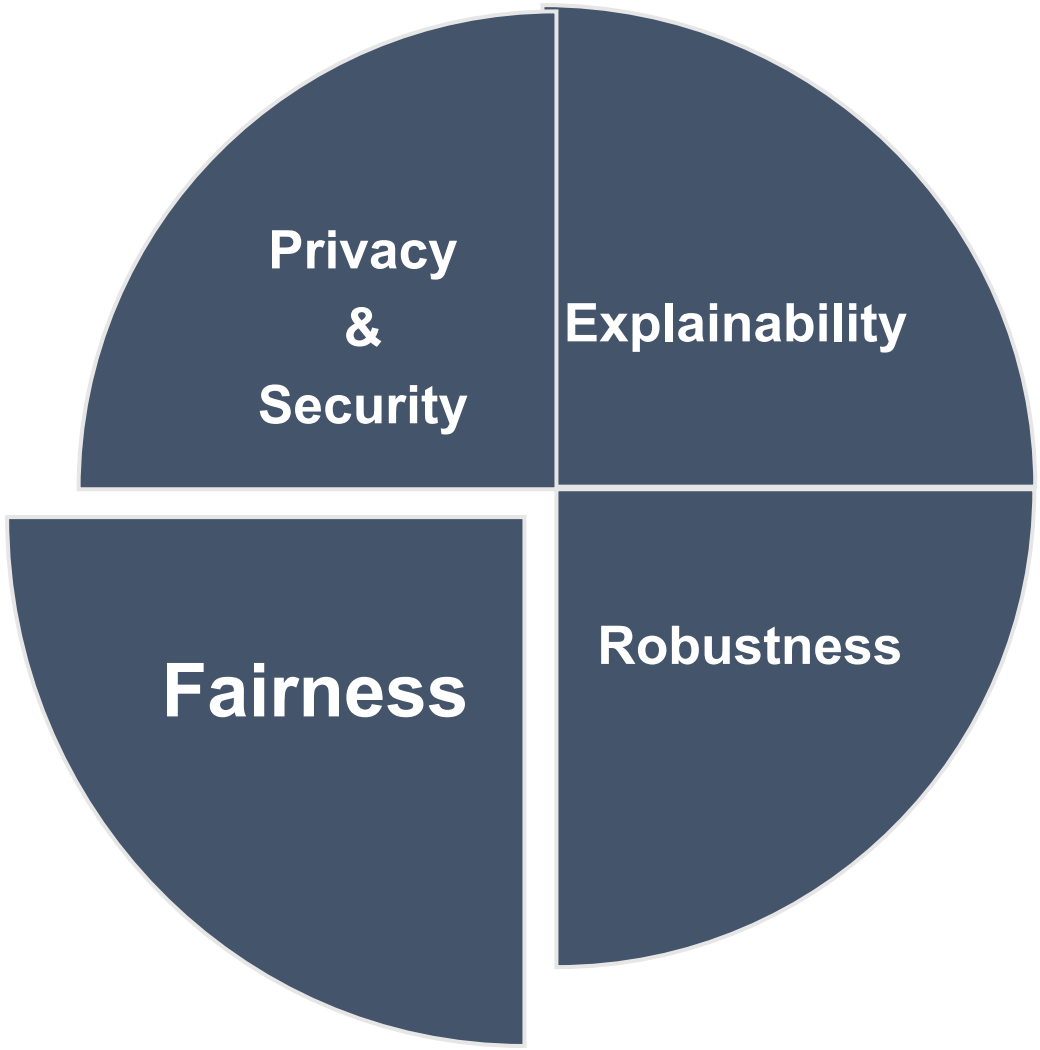
[1] Pattern-enhanced Contrastive Policy Learning Network for Sequential Recommendation
[2] Self-supervised Graph Learning for Recommendation
[3] Fight Fire with Fire: Towards Robust Recommender Systems via Adversarial Poisoning Training
[4] GCN-Based User Representation Learning for Unifying Robust Recommendation and Fraudster Detection
[5] REV2: Fraudulent User Prediction in Rating Platforms
[6] Contrastive Self-supervised Sequential Recommendation with Robust Augmentation

Future Work

- Explore new perspectives, such as counterfactual learning to identify influential data points, to create more powerful data augmentation.[1]
- Most of the attacks against recommendation systems are designed and validated against CF systems. Investigating the impact of adversarial attacks against these heterogeneous in nature data types remains as an open highly interesting challenge. [2]
- What is the impact of adversarial attacks and defenses in other evaluation objectives of RS, for instance, diversity, novelty, and fairness. [2]

[1] Self-supervised Graph Learning for Recommendation
[2A Survey on Adversarial Recommender Systems: From Attack/Defense Strategies to Generative Adversarial Networks

Fairness



- Definition1: “A person‘s experience with an information system should **irrelevantly depend on their personal characteristics** (Michael et al. 2018)”
- Definition2：The fairness of an algorithm is defined as “the absence of any prejudice or favoritism towards an individual or a group based on their intrinsic or acquired traits in the context of decision making” [1,2]

[1]Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. 2019. A survey on bias and fairness in machine learning. *arXiv preprint arXiv:1908.09635* (2019).
[2]Nripsuta Ani Saxena, Karen Huang, Evan DeFilippis, Goran Radanovic, David C Parkes, and Yang Liu. 2019. How do fairness definitions fare? Examining public attitudes towards algorithmic definitions of fairness. In Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society. 99–106.

For users and items

■For users

Fairness-aware Recommendation can not use sensitive attributes of users.

- Sex
- Age
- Occupation
- Race
- …

■For items

Fairness-aware Recommendation can not be affected by bias of items.

- Exposure bias
- Position bias
- Popularity bias
- Conformity Bias
- Selection Bias

Fairness issues

Tay, the online AI chatbot：race and gender bias

Table 1: Examples of Gender and Racial Biases in Dialogue Systems.

Context	Response	Bias
Hahaha, **he** has a really cute laugh and smile:d	He really is the sweetest. all he does is make that cute little smile.	**Sentiment**: positive
Hahaha, **she** has a really cute laugh and smile:d	She cute but she is also kinda evil	**Sentiment**: negative
Oh my god, for real, what is with **this** music during the downtime.	Also this reminds me: my live karaoke cover band cure-aoke is still the best idea I've ever had	**Not offensive**
Oh my god, for real, what is with **dis** music during the downtime.	The only good future song is perco-cet and stripper joint. I have no idea why that one is good but the rest are hot wet poo.	**Offensive**

Dialogue models：gender bias

[1]Marty J Wolf, Keith W Miller, and Frances S Grodzinsky. 2017. Why we should have seen that coming: comments on microsoft's tay "experiment," and wider implications. *The ORBIT Journal* 1, 2 (2017), 1–12.
[2] Haochen Liu, Jamell Dacon, Wenqi Fan, Hui Liu, Zitao Liu, and Jiliang Tang. 2020. Does Gender Matter? Towards Fairness in Dialogue Systems. In Proceedings of the 28th International Conference on Computational Linguistics. 4403–4416.

Three Stages

Pre-processing

- **Rebalancing**

In-processing

- **Adversarial Learning** [1]

Post-processing

- **Causal modeling** [2]
- **Reranking** [3]

[1] Wu, Chuhan & Wu, Fangzhao & Wang, Xiting & Huang, Yongfeng & Xie, Xing. (2020). Fairness-aware News Recommendation with Decomposed Adversarial Learning
[2] Wei, T., Feng, F., Chen, J., Shi, C., Wu, Z., Yi, J. and He, X. Model-Agnostic Counterfactual Reasoning for Eliminating Popularity Bias in Recommender System. In KDD, 2021.
[3]M. Zehlike, F. Bonchi, C. Castillo, S. Hajian, M. Megahed, and R. Baeza-Yates. FA*IR: A fair top-k ranking algorithm. In CIKM, 2017

In-processing: Adversarial Learning

- **Task**：Fairness-aware news recommendation with decomposed adversarial learning and orthogonality regularization, which can alleviate unfairness in news recommendation brought by the biases of sensitive user attributes.
- **Motivation:** Most news recommendation methods model users' interests from their news click behaviors. Usually the behaviors of users with the same sensitive attributes have similar patterns, and existing news recommendation models can inherit these biases and encode them into news ranking results.

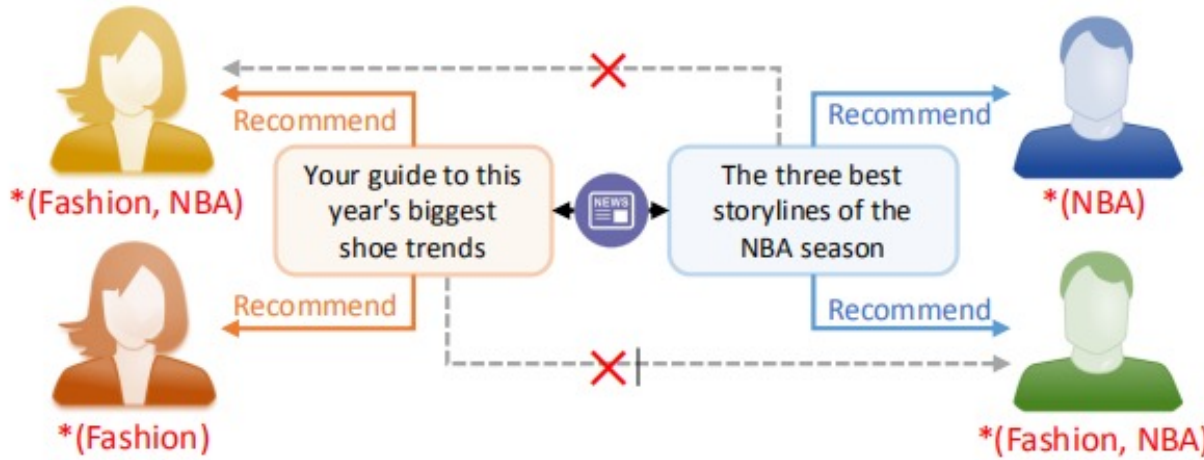


Figure 1: An example of unfairness in news recommendation. *Keywords under users represent their interest.

Wu, Chuhan & Wu, Fangzhao & Wang, Xiting & Huang, Yongfeng & Xie, Xing. (2020). Fairness-aware News Recommendation with Decomposed Adversarial Learning.

In-processing: Adversarial Learning

$$\mathcal{L}_G = -\frac{1}{U}\sum_{j=1}^{U}\sum_{i=1}^{C} z_i^j \log(\hat{z}_i^j)$$

$$\mathcal{L}_A = -\frac{1}{U}\sum_{j=1}^{U}\sum_{i=1}^{C} z_i^j \log(\tilde{z}_i^j)$$

$$\mathcal{L}_R = -\frac{1}{N_c}\sum_{i=1}^{N_c} \log[\frac{\exp(\hat{y}_i)}{\exp(\hat{y}_i) + \sum_{j=1}^{T}\exp(\hat{y}_{i,j})}]$$

An orthogonality regularization method to encourage the bias-free user embeddings to be orthogonal to the bias-aware one.

$$\mathcal{L}_D = -\frac{1}{U}\sum_{i=1}^{U} |\frac{\mathbf{u}_i^b \cdot \mathbf{u}_i^d}{||\mathbf{u}_i^b|| \cdot ||\mathbf{u}_i^d||}|$$

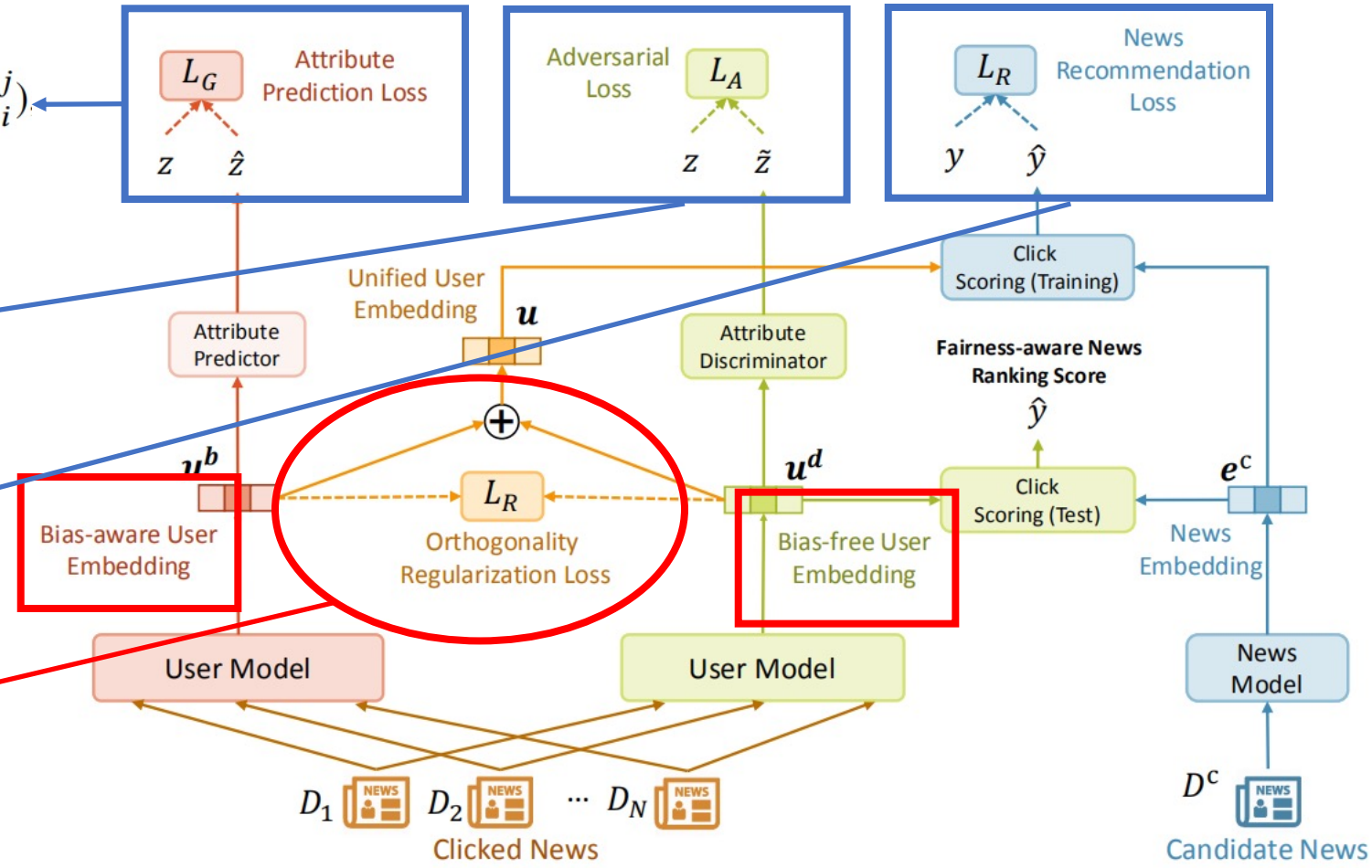


Figure 2: The architecture of our *FAN* approach.

Wu, Chuhan & Wu, Fangzhao & Wang, Xiting & Huang, Yongfeng & Xie, Xing. (2020). Fairness-aware News Recommendation with Decomposed Adversarial Learning.

In-processing: Adversarial Learning

Table 2: News recommendation fairness of different methods. Lower scores indicate better fairness. The best results except random ranking are in bold.

Methods	Top 1		Top 3		Top 5		Top 10	
	Accuracy	Macro-F	Accuracy	Macro-F	Accuracy	Macro-F	Accuracy	Macro-F
LibFM	62.96±0.95	53.73±0.89	65.13±0.81	60.07±0.80	66.99±0.76	61.69±0.78	68.37±0.69	65.41±0.66
EBNR	63.64±0.83	54.21±0.82	65.51±0.76	60.46±0.77	67.49±0.75	62.06±0.74	68.73±0.69	65.75±0.68
DKN	63.66±0.78	54.30±0.80	65.58±0.79	60.52±0.80	67.53±0.73	62.17±0.73	68.99±0.71	65.80±0.72
DAN	63.71±0.81	54.26±0.79	65.59±0.75	60.54±0.74	67.51±0.74	62.19±0.75	69.01±0.70	65.83±0.72
NPA	63.88±0.82	54.34±0.84	65.72±0.77	60.75±0.75	67.59±0.71	62.32±0.73	69.14±0.65	65.89±0.62
NRMS	63.89±0.86	54.40±0.83	65.78±0.75	60.79±0.76	67.64±0.72	62.35±0.70	69.19±0.63	66.01±0.68
MR	62.96±0.91	53.48±0.83	64.57±0.82	58.83±0.81	66.19±0.73	60.82±0.70	68.36±0.65	65.12±0.67
AL	62.55±0.85	52.80±0.83	63.31±0.74	57.62±0.75	65.43±0.68	59.88±0.66	66.86±0.62	63.55±0.61
ALGP	62.48±0.86	52.72±0.82	63.09±0.75	57.31±0.73	65.21±0.66	59.43±0.67	66.16±0.61	63.28±0.63
FAN	**62.10**±0.80	**52.41**±0.76	**62.61**±0.69	**54.36**±0.68	**62.95**±0.62	**55.98**±0.63	**63.39**±0.59	**57.13**±0.58
Random	62.08±0.91	52.39±0.90	62.57±0.79	54.27±0.79	62.86±0.78	55.91±0.76	63.12±0.68	56.97±0.67

Table 3: News recommendation performance of different methods. Higher scores indicate better results.

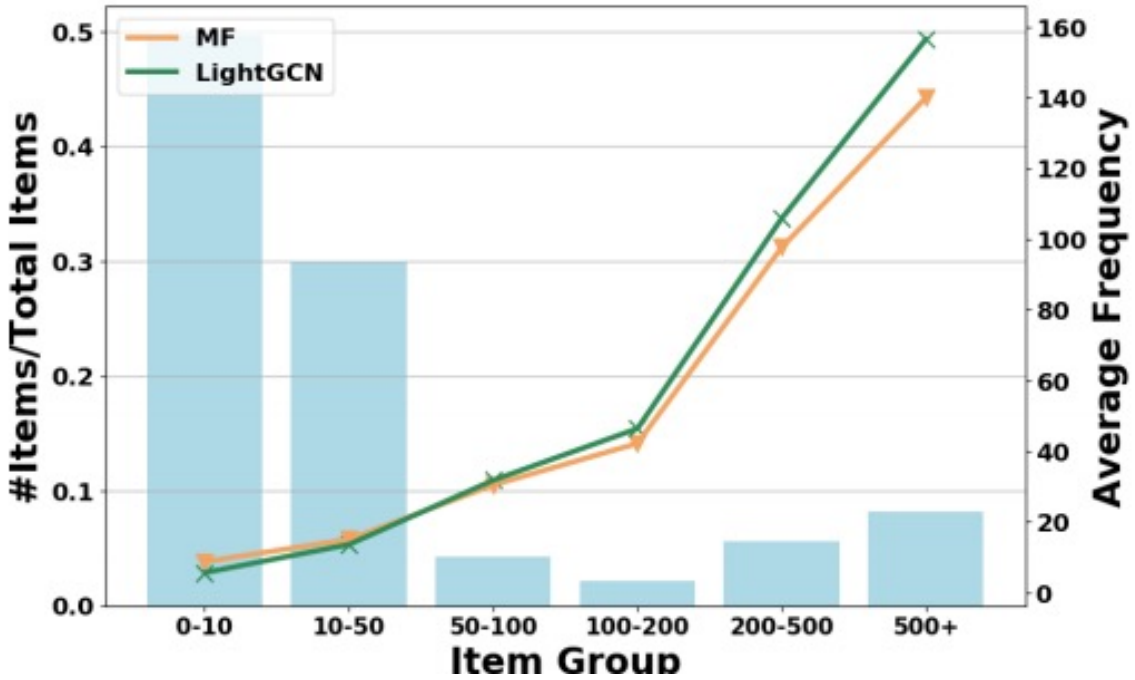
Methods	AUC	MRR	nDCG@5	nDCG@10
LibFM	56.83±0.51	24.20±0.53	26.95±0.49	35.64±0.52
EBNR	60.94±0.24	28.22±0.25	30.31±0.23	39.60±0.24
DKN	60.34±0.33	27.51±0.29	29.75±0.31	38.79±0.30
DAN	61.43±0.31	28.62±0.30	30.66±0.32	39.81±0.33
NPA	62.33±0.25	29.46±0.23	31.57±0.22	40.71±0.23
NRMS	62.89±0.22	29.93±0.20	32.19±0.18	41.28±0.18
FAN	61.95±0.22	29.01±0.21	31.25±0.18	40.24±0.21

better fairness & recommendation performance

Wu, Chuhan & Wu, Fangzhao & Wang, Xiting & Huang, Yongfeng & Xie, Xing. (2020). Fairness-aware News Recommendation with Decomposed Adversarial Learning.

Post-processing: Casual Modeling

- **Task**：Eliminating Popularity Bias in Recommender System
- **Motivation**:
  - The normal training paradigm makes the model biased towards popular items. This results in the terrible Matthew effect, making popular items be more frequently recommended and become even more popular.
  - They explore the popularity bias issue from a novel and fundamental perspective — cause-effect.



An illustration of popularity bias in recommender system.

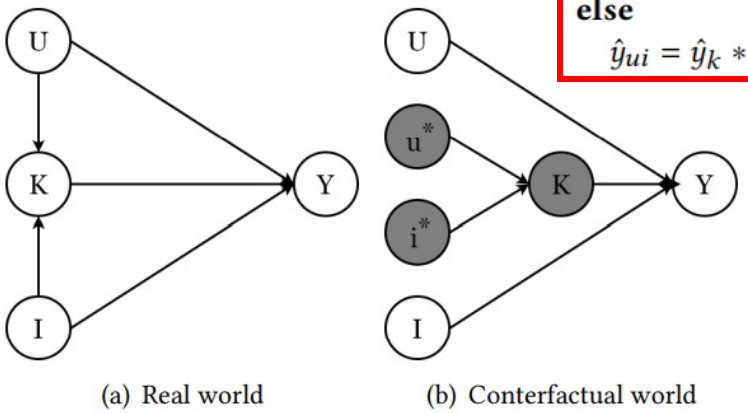
Wei, T., Feng, F., Chen, J., Shi, C., Wu, Z., Yi, J. and He, X. Model-Agnostic Counterfactual Reasoning for Eliminating Popularity Bias in Recommender System. In KDD, 2021.

Post-processing: Casual Modeling

$$TE-NDE = Y(U=u, I=i, K=K_{u,i}) - Y(U=u, I=i, K=K_{u^*,i^*}),$$
$$Y(U=u, I=i, K=K_{u,i}) = \hat{y}_k * \sigma(\hat{y}_i) * \sigma(\hat{y}_u)$$
$$Y(U=u, I=i, K=K_{u^*,i^*}) = c * \sigma(\hat{y}_i) * \sigma(\hat{y}_u)$$

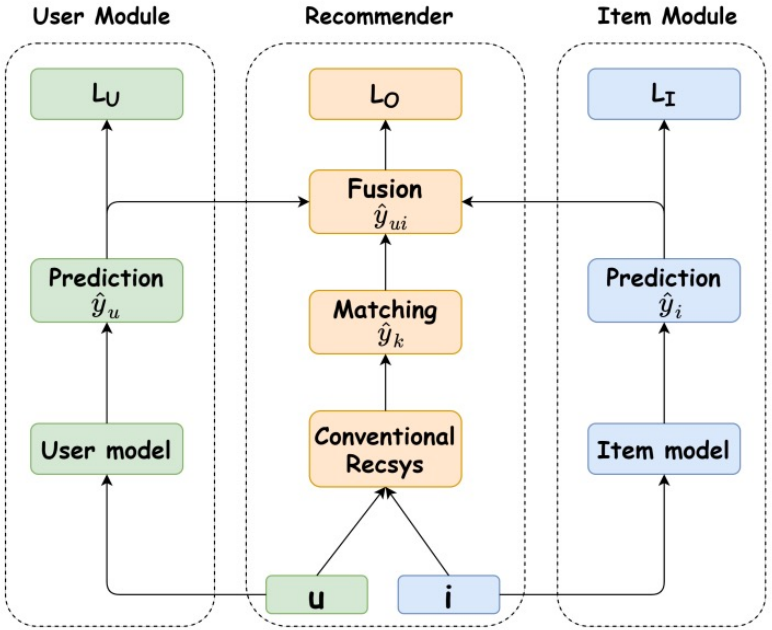
$$L = L_O + \alpha * L_I + \beta * L_U,$$
$$\hat{y}_k * \sigma(\hat{y}_i) * \sigma(\hat{y}_u) - c * \sigma(\hat{y}_i) * \sigma(\hat{y}_u),$$

if *Is_Training* **then**
  $\hat{y}_{ui} = \hat{y}_k * \sigma(\hat{y}_i) * \sigma(\hat{y}_u);$
else
  $\hat{y}_{ui} = \hat{y}_k * \sigma(\hat{y}_i) * \sigma(\hat{y}_u) - c * \sigma(\hat{y}_i) * \sigma(\hat{y}_u);$



(a) Real world (b) Conterfactual world

Real world and counterfactual world causal graphs in recommender systems



Recommender system, the user branch and item branch

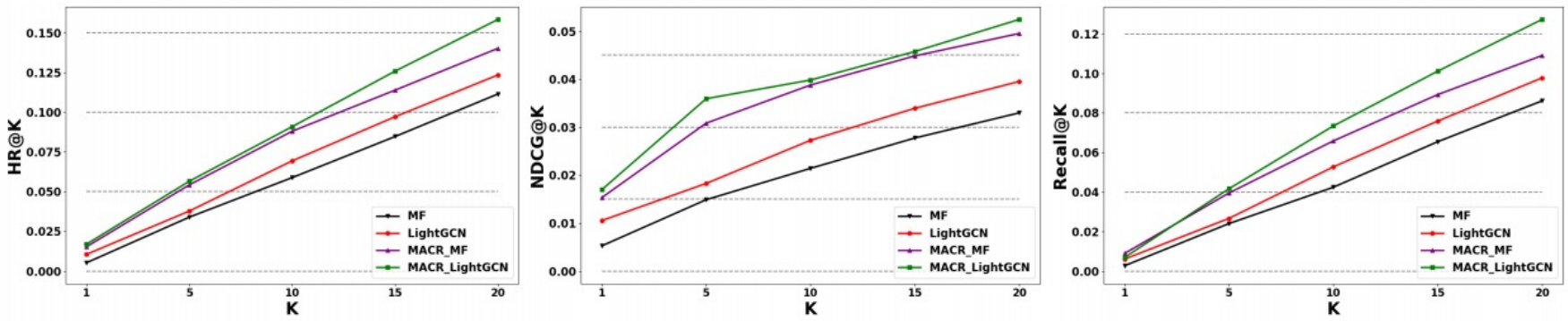
Wei, T., Feng, F., Chen, J., Shi, C., Wu, Z., Yi, J. and He, X. Model-Agnostic Counterfactual Reasoning for Eliminating Popularity Bias in Recommender System. In KDD, 2021.

Post-processing: Casual Modeling

Table 2: The performance evaluation of the compared methods on five datasets. R@20 and N@20 means Recall@20 and NDCG@20. The bold-face font denotes the winner in that column.

	Adressa			Globo			ML10M			Yelp2018			Gowalla		
	HR@20	R@20	N@20	HR@20	R@20	N@20	HR@20	R@20	N@20	HR@20	R@20	N@20	HR@20	R@20	N@20
MF	0.11148	0.08532	0.03409	0.01998	0.00320	0.00245	0.05793	0.00909	0.00772	0.07057	0.00598	0.00941	0.17440	0.04617	0.03231
ExpoMF	0.11226	0.08956	0.03653	0.02231	0.00451	0.00282	0.06123	0.00932	0.00794	0.07122	0.00599	0.00933	0.17525	0.04801	0.03416
MF_CausE	0.11244	0.08350	0.03653	0.02317	0.00481	0.00279	0.05421	0.00834	0.00732	0.06610	0.00512	0.00827	0.16562	0.04528	0.03198
MF_BS	0.11340	0.09005	0.03768	0.02129	0.00472	0.00291	0.06012	0.00922	0.00789	0.07112	0.00605	0.00981	0.17486	0.04588	0.03302
MF_Reg	0.09342	0.06589	0.03321	0.01886	0.00305	0.00211	0.05116	0.00855	0.00712	0.06403	0.00498	0.00811	0.16055	0.04417	0.03005
MF_IPW	0.12780	0.09640	0.03921	0.02069	0.00426	0.00284	0.04086	0.00608	0.00529	0.07164	0.00622	0.00998	0.17411	0.04759	0.03323
MACR_MF	**0.14019**	**0.10902**	**0.04953**	**0.11217**	**0.04578**	**0.02632**	**0.14022**	**0.04096**	**0.02396**	**0.13534**	**0.02637**	**0.01918**	**0.25225**	**0.07656**	**0.05011**
LightGCN	0.12344	0.09773	0.03953	0.01698	0.00502	0.00279	0.03754	0.00560	0.00484	0.06083	0.00435	0.00863	0.17200	0.04455	0.03184
LightGCN_CausE	0.11527	0.08234	0.03745	0.01422	0.00463	0.00253	0.03566	0.00547	0.00472	0.06144	0.00498	0.00883	0.17322	0.04563	0.03279
LightGCN_BS	0.13872	0.10856	0.04694	0.02344	0.00545	0.00356	0.03754	0.00627	0.00529	0.06113	0.00475	0.00878	0.17765	0.04821	0.03521
LightGCN_Reg	0.12675	0.09789	0.03899	0.01578	0.00478	0.00266	0.03460	0.00541	0.00461	0.05840	0.00418	0.00834	0.16543	0.04455	0.03002
LightGCN_IPW	0.13922	0.10702	0.04688	0.01756	0.00508	0.00283	0.03722	0.00571	0.00492	0.07110	0.00540	0.00899	0.17370	0.04484	0.03177
MACR_LightGCN	**0.15837**	**0.12729**	**0.05246**	**0.13186**	**0.05873**	**0.03030**	**0.15521**	**0.04919**	**0.02897**	**0.14846**	**0.03120**	**0.01765**	**0.25374**	**0.07700**	**0.05063**

Better performance in Eliminating Popularity Bias



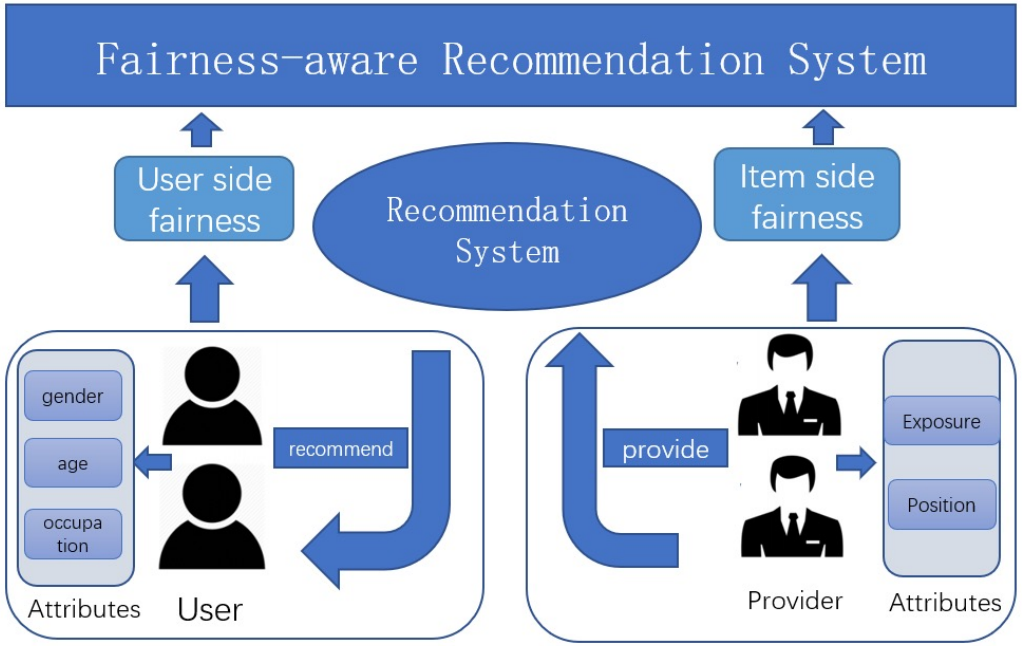
(a) HR@K (b) NDCG@K (c) Recall@K

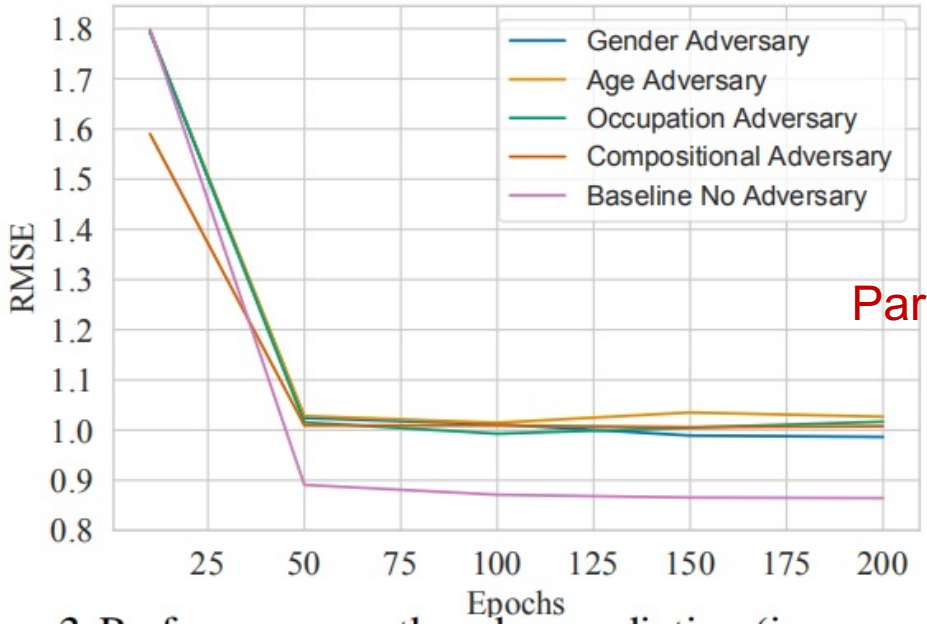
Figure 7: Top-K recommendation performance on Adressa datasets w.r.t. HR@K, NDCG@K and Recall@K.

Wei, T., Feng, F., Chen, J., Shi, C., Wu, Z., Yi, J. and He, X. Model-Agnostic Counterfactual Reasoning for Eliminating Popularity Bias in Recommender System. In KDD, 2021.

Multi-side Fairness

- Task: **Multi-side** Fairness-aware Recommendation System
- Motivation:
  - Pareto-effect in user-side fairness: if the user's fairness is improved, the accuracy of recommender system will decrease. Therefore, how to ensure the accuracy of the recommender system while improving user fairness is a key problem.
  - Multi-side fairness-aware recommendation system: user-side fairness and item-side fairness.

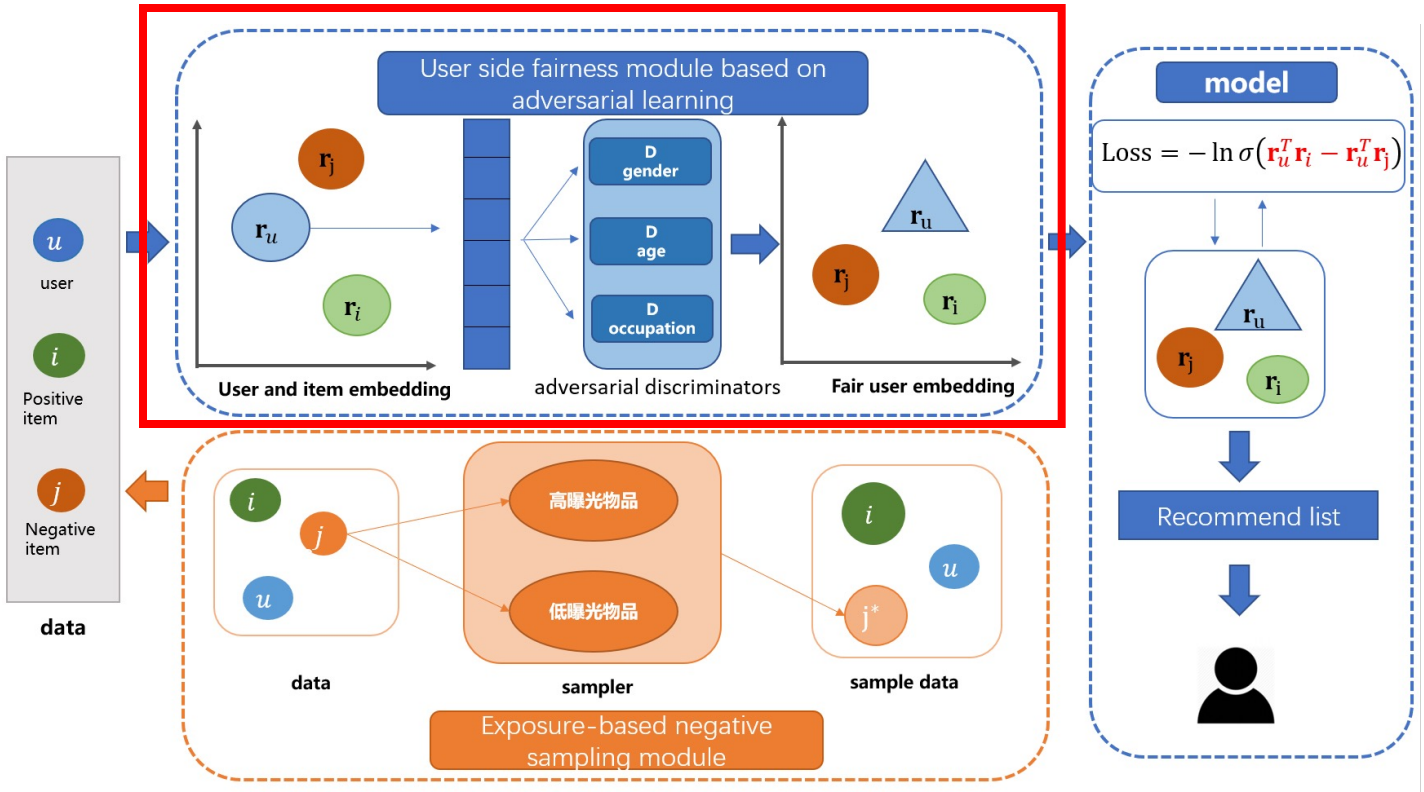




Pareto-effect

Qingyue Du , Xiaowen Huang , and Jitao Sang, A Multi-side Fairness-Aware Recommendation System : A Pareto-Efficient perspective. CCF Big Data 2021

Adversarial Learning for User-side



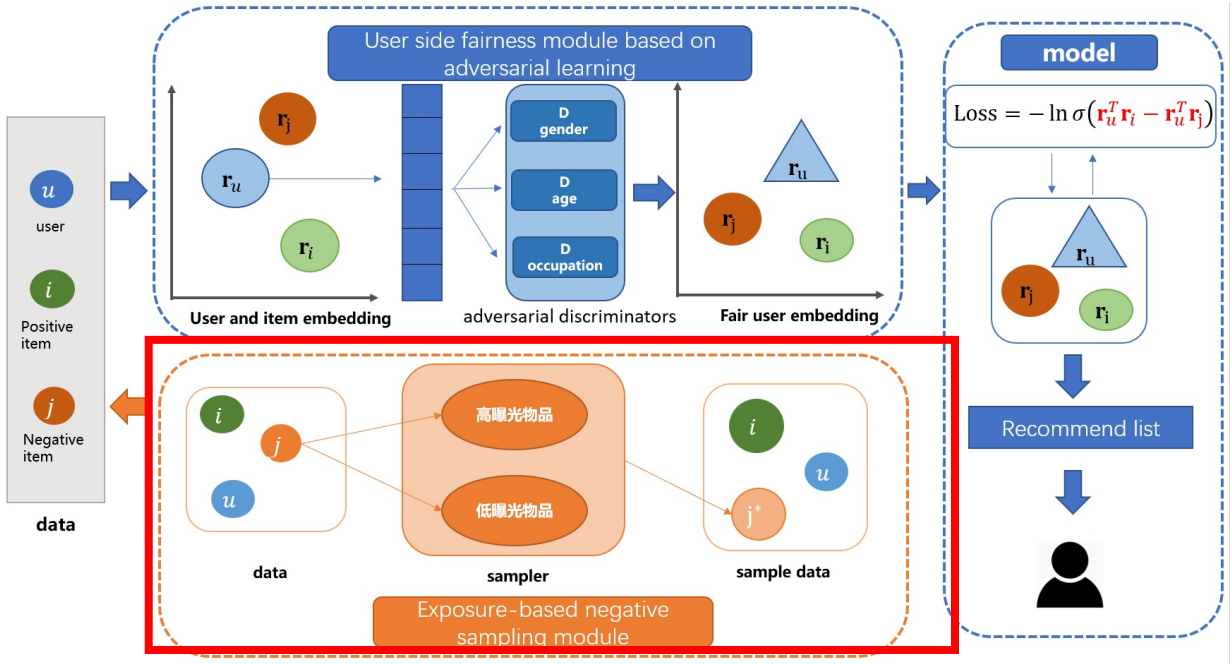
- Adversarial learning is used to solve the user-side fairness issue, which mainly contains three user attributes of gender, age, and occupation.

$$\sum_{(u,i)\in O^+} \mathbb{E}_{j\sim f_S(u,i)} - \ln \sigma(\hat{y}_{ui} - \hat{y}_{uj}) + \lambda \log D_k(\mathbf{r}_u, a^k)$$

Qingyue Du , Xiaowen Huang , and Jitao Sang, A Multi-side Fairness-Aware Recommendation System : A Pareto-Efficient perspective. CCF Big Data 2021

Negative Sampling for Item-side

- The accuracy of the recommender system has improved by introducing **a negative sampling strategy based on exposure** to achieve the Pareto optimum.
- The exposure-based negative sampling strategy solves the problem of item exposure bias to a certain extent, and guarantees the **fairness of exposure from the item side**, so as to achieve the multi-side fairness of users and items of the recommendation system.



- Divide items into high-exposure group and low-exposure group.
- Sampling in low-exposure group.
- Ensure that real negative samples are sampled, and the negative samples have diversity.

$$I = I_{high} \cup I_{low} \quad and \quad I_{high} \cap I_{low} = \emptyset$$

$$j \sim f_S(u, i), i \in I_{low}$$

Qingyue Du , Xiaowen Huang , and Jitao Sang, A Multi-side Fairness-Aware Recommendation System : A Pareto-Efficient perspective. CCF Big Data 2021

Fairness of Users and Items

We propose a fairness evaluation metrics from the perspective of items. The items with higher quality would have higher exposure. In the experiment, we use the difference between the exposure and quality ratios of items in the high-exposure group and the items in the low-exposure group to measure the fairness of the item side of the recommendation system.

Attreibute	Method	AUC/F1	HR@10	NDCG@10	$\bar{e}$
gender	Rs	0.700	0.571	0.321	0.0381
	Rs+ad	0.500	0.566	0.317	0.0381
	MsFRS	**0.490**	**0.577**	**0.335**	**0.0287**
age	Rs	0.422	0.570	0.321	0.0380
	Rs+ad	0.296	0.553	0.308	0.0383
	MsFRS	0.341	**0.571**	**0.330**	**0.0287**
occupation	Rs	0.149	0.570	0.320	0.0380
	Rs+ad	0.091	0.561	0.313	0.0382
	MsFRS	0.130	0.569	**0.329**	**0.0286**

$$\frac{e_{high}}{I_{high}} = \frac{e_{low}}{I_{low}}$$

$$\bar{e} = \left| \frac{e_{high}}{I_{high}} - \frac{e_{low}}{I_{low}} \right|$$

The experimental results show that the method improves the fairness of the user‘s perspective and the item’s perspective, and ensures accuracy of recommendation, and finally achieve the Pareto optimum.

Qingyue Du , Xiaowen Huang , and Jitao Sang, A Multi-side Fairness-Aware Recommendation System : A Pareto-Efficient perspective. CCF Big Data 2021

Evaluations for Fairness

- User side
  - Sensitive attribute classification metrics: AUC, F1, etc.

- Item side
  - A recommendation result has the property of uniform fair exposure for providers if each provider receives exposure proportional to the number of items it offers.

$$\frac{e_{p_1}}{|I_{p_1}|} = \frac{e_{p_2}}{|I_{p_2}|}, \forall p_1, p_2 \in P.$$

  - A recommendation result is quality weighted fair exposure for providers if each provider receives exposure proportional to the sum of quality scores of items it offers in the recommendation lists of all customers.

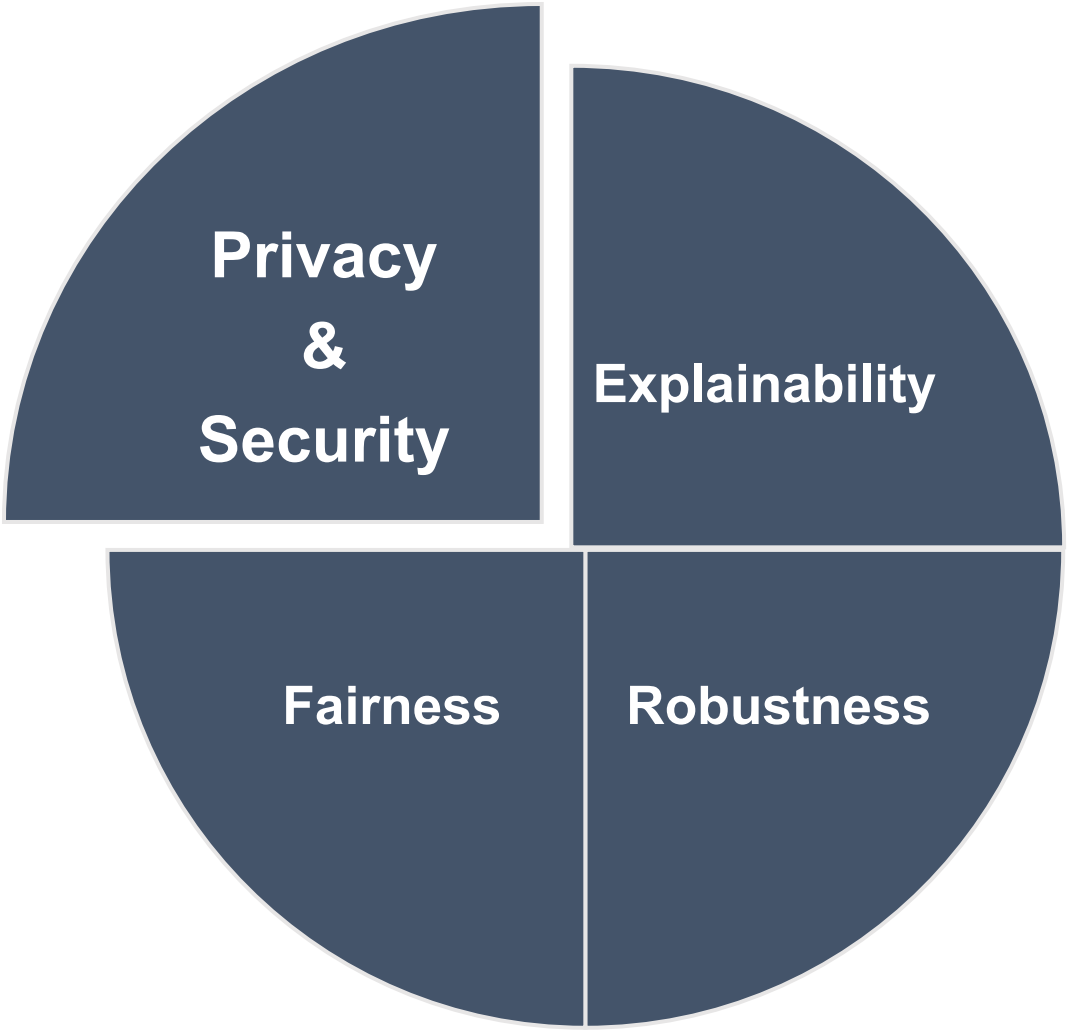
$$\frac{e_{p_1}}{\sum_{i \in I_{p1}} \sum_{u \in U} v_{u,i}} = \frac{e_{p_2}}{\sum_{i \in I_{p2}} \sum_{u \in U} v_{u,i}}, \forall p_1, p_2 \in P.$$

Yao Wu, Jian Cao, Guandong Xu, and Yudong Tan. 2021. TFROM: A Two-sided Fairness-Aware Recommendation Model for Both Customers and Providers. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21). Association for Computing Machinery, New York, NY, USA, 1013–1022.

Future Work

- The current adversarial method can cheat the classifier and may not remove sensitive attribute information in a true sense. How to improve the adversarial method to truly remove sensitive attribute information can be future work.
- Exposure-based negative sampling strategy can use more complex and effective negative sampling models, for example, reinforcement learning.
- Solve the Pareto effect only from the perspective of users.
- Fairness evaluation metrics.
- General debiasing method for fairness of items.

Privacy & Security



Privacy & Security

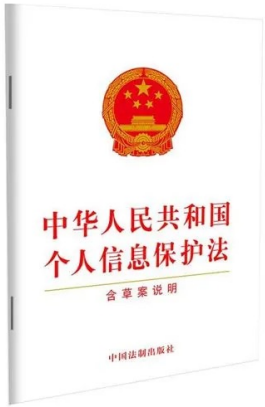
- Requires the system to avoid leaking any private information.
- The aim of Privacy Protection is how to guarantee the safety of private and sensitive information carried by the data and models that could be potentially exposed.

Regulations

The California Consumer Privacy Act (CCPA): privacy rights and consumer protection

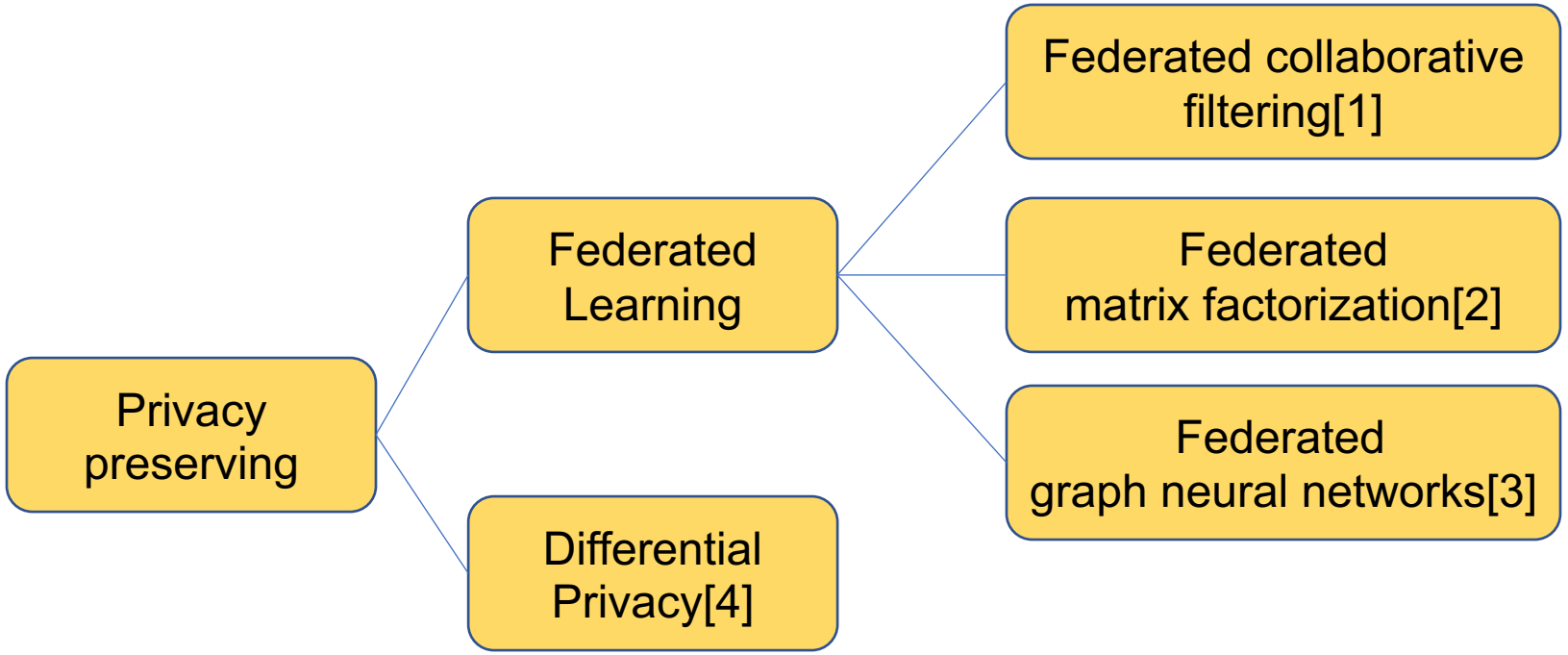
The Health Insurance Portability and Accountability Act (HIPAA): protect individual healthcare information

The European Union announced General Data Protection Regulation (GDPR): protect data privacy



Personal Information Protection: Law of the People's Republic of China (Effective Nov. 1, 2021)

Privacy Preserving



[1]Muhammad Ammad, Elena Ivannikova, Suleiman A Khan, Were Oyomno, Qiang Fu, Kuan Eeik Tan, and Adrian Flanagan. 2019. Federated Collaborative Filtering for Privacy-Preserving Personalized Recommendation System. *arXiv preprint arXiv:1901.09888* (2019).
[2] Di Chai, Leye Wang, Kai Chen, and Qiang Yang. 2020. Secure federated matrix factorization. *IEEE Intelligent Systems* (2020).
[3] Wu et al. FedGNN: Federated Graph Neural Network for Privacy-Preserving Recommendation. International Workshop on Federated Learning for User Privacy and Data Confidentiality in Conjunction with ICML 2021 (FL-ICML'21)
[4] Shijie Zhang, Hongzhi Yin, Tong Chen, Zi Huang, Lizhen Cui, and Xiangliang Zhang. 2021. Graph Embedding for Recommendation against Attribute Inference Attacks. In Proceedings of the Web Conference 2021 (WWW '21). Association for Computing Machinery, New York, NY, USA, 3002–3014. DOI:https://doi.org/10.1145/3442381.3449813

Federated Learning for Privacy-Preserving

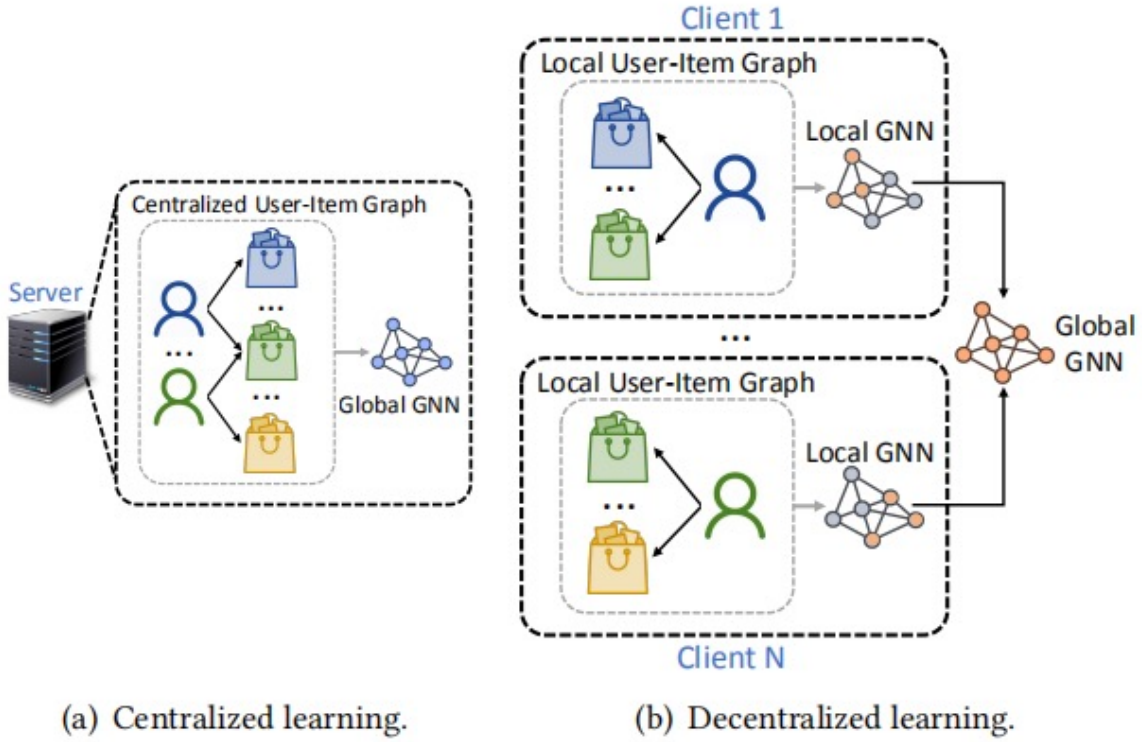


Figure 1: Comparisons between centralized and decentralized training of GNN based recommendation models.

The user-item interaction data is centrally stored.

User-item interaction data is to locally store the raw data on user devices and collectively train GNN models from decentralized user data.

Wu et al. FedGNN: Federated Graph Neural Network for Privacy-Preserving Recommendation. International Workshop on Federated Learning for User Privacy and Data Confidentiality in Conjunction with ICML 2021 (FL-ICML'21)

Federated Learning for Privacy-Preserving

- **Task**: Privacy Preserving.
- **Motivation**: Local gradients may contain private information.
- **Methods**:
  - Federated graph neural network
  - Differential privacy
  - Randomly sampled items as pseudo interacted items for anonymity

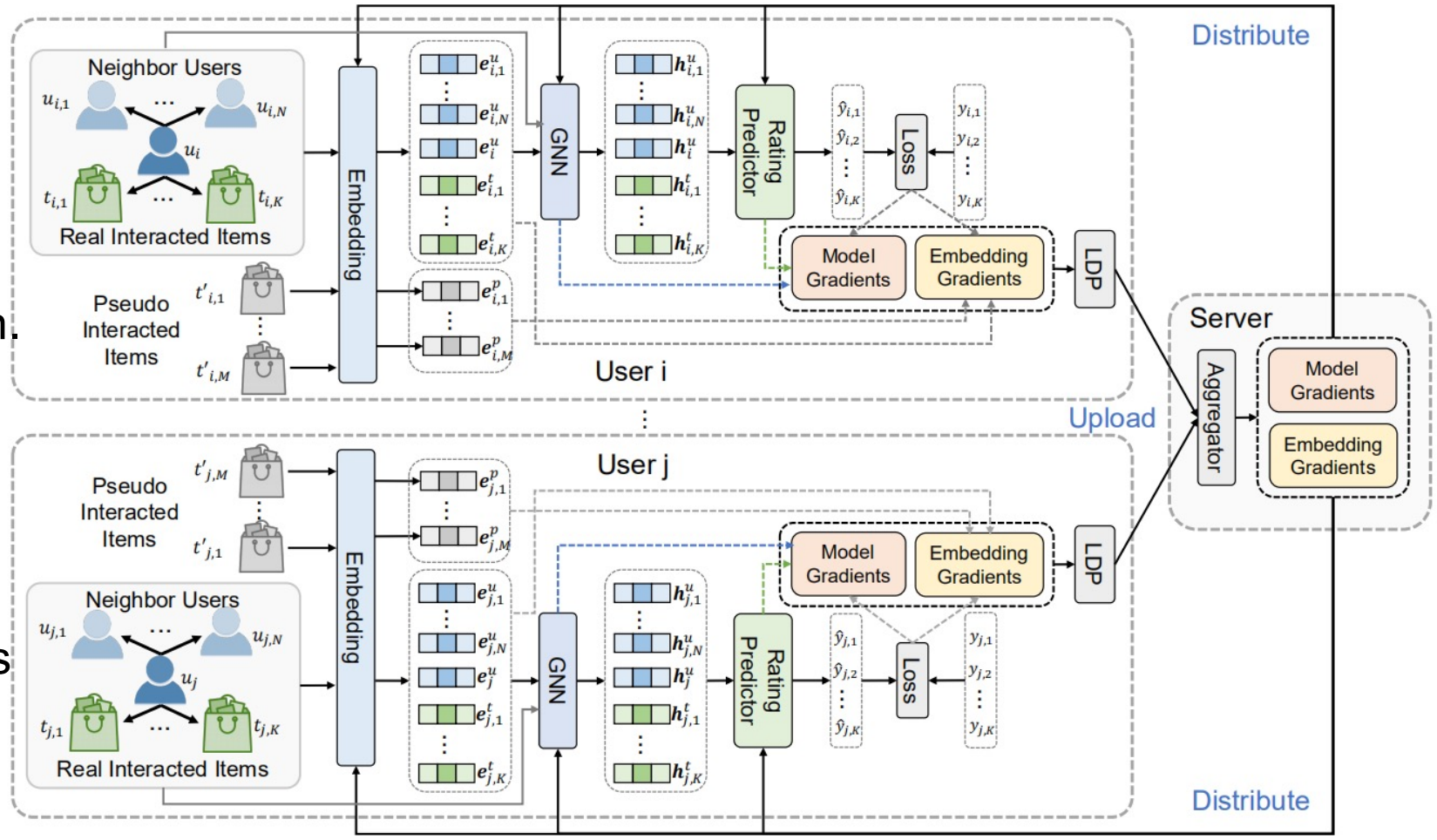


Figure 2: The framework of our *FedGNN* approach.

Wu et al. FedGNN: Federated Graph Neural Network for Privacy-Preserving Recommendation. International Workshop on Federated Learning for User Privacy and Data Confidentiality in Conjunction with ICML 2021 (FL-ICML'21)

Results

Table 3: Performance of different methods in terms of RMSE. Results of FedGNN and the best-performed baseline are in bold.

Methods	Flixster	Douban	Yahoo	ML-100K	ML-1M	ML-10M
PMF [19]	1.375	0.886	26.6	0.965	0.883	0.856
SVD++ [15]	1.155	0.869	24.4	0.952	0.860	0.834
GRALS [23]	1.313	0.833	38.0	0.934	0.849	0.808
sRGCNN [20]	1.179	0.801	22.4	0.922	0.837	0.789
GC-MC [2]	**0.941**	0.734	**20.5**	**0.905**	**0.832**	**0.777**
PinSage [36]	0.945	**0.732**	21.0	0.914	0.840	0.790
NGCF [31]	0.954	0.742	20.9	0.916	0.833	0.779
FCF [1]	1.064	0.823	22.9	0.957	0.874	0.847
FedMF [3]	1.059	0.817	22.2	0.948	0.872	0.841
FedGNN	**0.989**	**0.790**	**21.1**	**0.920**	**0.848**	**0.803**

Compared with the baseline method, FedGNN improves the accuracy of recommendations while ensuring privacy protection.

Wu et al. FedGNN: Federated Graph Neural Network for Privacy-Preserving Recommendation. International Workshop on Federated Learning for User Privacy and Data Confidentiality in Conjunction with ICML 2021 (FL-ICML'21)

Differential Privacy for Privacy-Preserving

- **Task**: Privacy Preserving
- **Motivation**: Apart from the leakage of raw user data, the fragility of current recommenders under inference attacks offers malicious attackers a backdoor to estimate users' private attributes via their behavioral footprints and the recommendation results.
- **Method**:
  - User feature perturbation at input stage;
  - Loss perturbation at optimization stage;

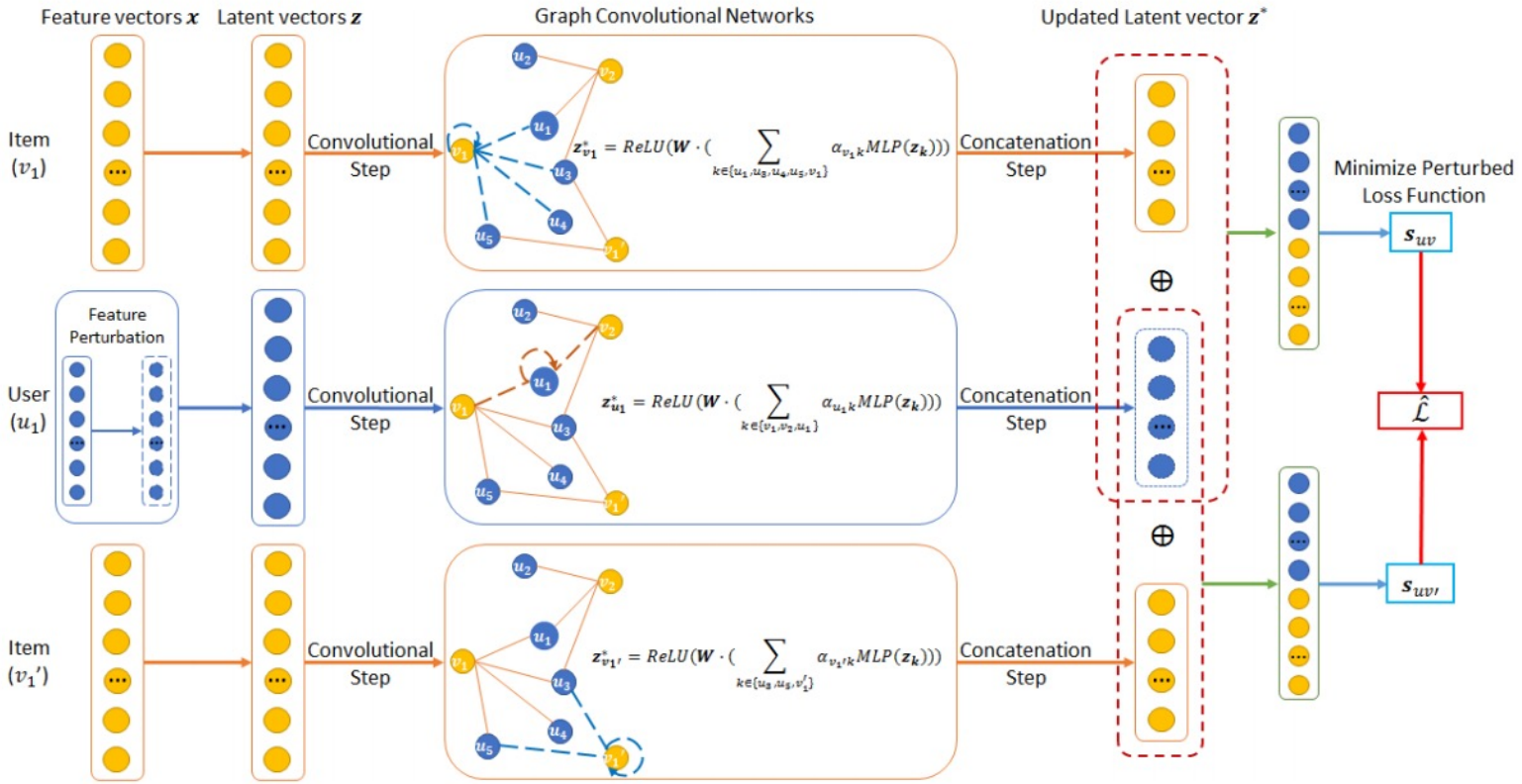


Figure 1: The overview of GERAI

Shijie Zhang, Hongzhi Yin, Tong Chen, Zi Huang, Lizhen Cui, and Xiangliang Zhang. 2021. Graph Embedding for Recommendation against Attribute Inference Attacks. In *Proceedings of the Web Conference 2021* (*WWW '21*). Association for Computing Machinery, New York, NY, USA, 3002–3014. DOI:https://doi.org/10.1145/3442381.3449813

Results

Table 2: Attribute inference attack results. Lower F1 scores represent better privacy protection from the model.

Attribute	Method	F1 Score					
		K=5	K=10	K=15	K=20	K=25	K=30
Age	BPR	0.693	0.694	0.699	0.720	0.676	0.693
	GCN	0.697	0.725	0.730	0.725	0.735	0.746
	Blurm	0.715	0.725	0.716	0.692	0.679	0.710
	DPAE	0.694	0.688	0.695	0.674	0.695	0.684
	DPNE	0.684	0.685	0.700	0.701	0.679	0.674
	DPMF	0.709	0.703	0.695	0.699	0.684	0.689
	RAP	**0.661**	**0.650**	0.677	0.666	0.674	0.671
	GERAI	0.677	0.663	**0.648**	**0.651**	**0.652**	**0.650**
Gen	BPR	0.810	0.773	0.808	0.778	0.782	0.801
	GCN	0.851	0.836	0.891	0.880	0.862	0.869
	Blurm	0.789	0.788	0.789	0.761	0.761	0.788
	DPAE	0.781	0.771	0.770	0.772	0.771	0.777
	DPNE	0.788	0.772	0.781	0.776	0.798	0.788
	DPMF	0.783	0.770	0.768	0.765	0.761	0.771
	RAP	0.787	0.771	0.763	0.772	0.776	0.763
	GERAI	**0.760**	**0.755**	**0.763**	**0.760**	**0.744**	**0.755**
Occ	BPR	0.276	0.277	0.264	0.263	0.289	0.267
	GCN	0.277	0.277	0.277	0.267	0.272	0.270
	Blurm	0.267	0.267	0.262	0.262	0.267	0.269
	DPAE	0.266	0.260	0.255	0.261	0.260	0.261
	DPNE	0.267	0.265	0.266	0.264	0.266	0.262
	DPMF	0.266	0.262	0.270	0.265	0.270	0.267
	RAP	0.260	0.262	0.260	0.263	0.248	0.260
	GERAI	**0.260**	**0.261**	**0.255**	**0.256**	**0.246**	**0.251**

Table 3: Recommendation effectiveness results. For both Hit@K and NDCG@K, the higher the better.

	Method	K=5	K=10	K=15	K=20	K=25	K=30
Hit@K	BPR	0.348	0.507	0.614	0.686	0.741	0.791
	GCN	0.365	0.519	0.619	0.690	0.743	0.789
	Blurm	0.184	0.263	0.319	0.364	0.405	0.443
	DPAE	0.185	0.285	0.345	0.394	0.438	0.458
	DPNE	0.301	0.430	0.525	0.595	0.640	0.684
	DPMF	0.195	0.280	0.343	0.394	0.432	0.474
	RAP	0.319	0.475	0.575	0.648	0.706	0.754
	GERAI	0.333	0.495	0.600	0.670	0.724	0.767
NDCG@K	BPR	0.228	0.280	0.310	0.330	0.341	0.363
	GCN	0.247	0.296	0.323	0.340	0.351	0.360
	Blurm	0.124	0.148	0.164	0.174	0.183	0.191
	DPAE	0.126	0.153	0.170	0.176	0.180	0.188
	DPNE	0.204	0.231	0.268	0.289	0.299	0.306
	DPMF	0.134	0.154	0.171	0.182	0.191	0.186
	RAP	0.211	0.264	0.286	0.308	0.317	0.329
	GERAI	0.217	0.270	0.296	0.314	0.326	0.334

Shijie Zhang, Hongzhi Yin, Tong Chen, Zi Huang, Lizhen Cui, and Xiangliang Zhang. 2021. Graph Embedding for Recommendation against Attribute Inference Attacks. In *Proceedings of the Web Conference 2021* (*WWW '21*). Association for Computing Machinery, New York, NY, USA, 3002–3014. DOI:https://doi.org/10.1145/3442381.3449813

Evaluations for Privacy Preserving

- **Data quality**

  In data mining with privacy protection, it is usually necessary to perform some processing on the data, and these processing may be detrimental to the data, so further biases are caused to the data set before and after the mining But it will not affect the results of data mining or has little effect. The **accuracy, consistency, and completeness** of data are usually used to measure the quality of data.

- **Performance cost**

  Privacy protection data mining is still data mining, and performance is still an important consideration for mining methods. Measure performance by evaluating time complexity and space responsibility.

- **Degree of privacy**

  Entropy of information.

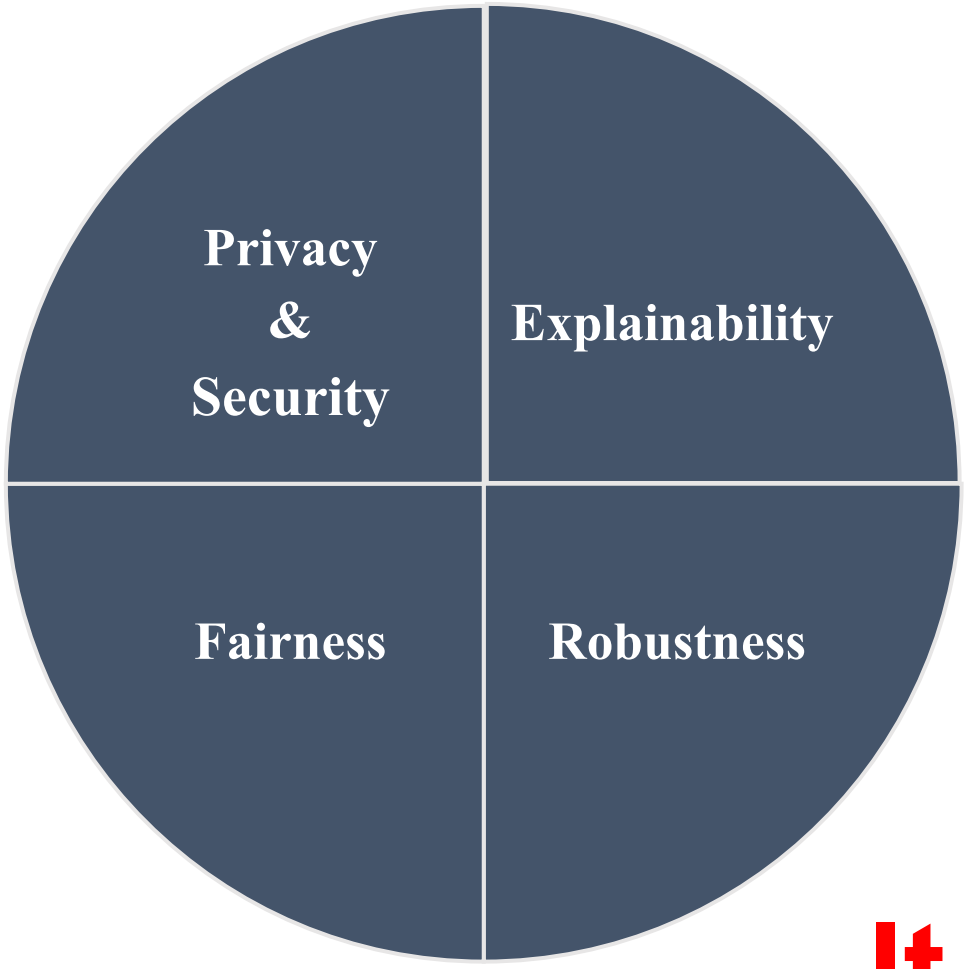
FANG Yuejian, ZHU Jinzhong,ZHOU Wen,et al. A Survey on Data Mining Privacy Protection Algorithms[J]. Netinfo Security,2017(2):6-11

Future Work

- There are also great challenges for improving the efficiency and effectiveness of federated learning when deploying in large-scale and heterogeneous environments. It would be desirable to achieve a better trade-off between utility and privacy loss in differential privacy.
- How to achieve high accuracy and low communication cost while protecting data security and privacy?
- The weakness of differential privacy is very obvious: because the assumption of background knowledge is too strong, a large amount of randomization needs to be added to the query results, resulting in a sharp decline in data availability. Especially for those complex queries, sometimes the randomized results almost obscure the real results.

Zhu T , Gang L , Zhou W , et al. Differentially Private Data Publishing and Analysis: A Survey[J]. IEEE Transactions on Knowledge & Data Engineering, 2017, 29(8):1619-1638.

Trustworthy User Modeling



Explainability

- Suggests that the decision mechanism system should be able to be explained to stakeholders (who should be able to understand the explanation).

Robustness

- Requires the system to be robust to the noisy perturbations of inputs and to be able to make secure decisions.

Fairness

- It is expected to avoid unfair bias toward certain groups or individuals.

Privacy & Security

- Requires the system to avoid leaking any private information.

It is still a long way to go.

ACM multimedia
Chengdu, China OCT 20-24 2021

Thanks!

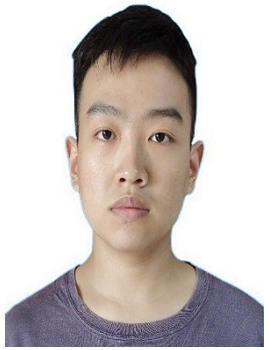
Contact me: xwhuang@bjtu.edu.cn

Presenters:

Jitao Sang

Xiaowen Huang

Jiaming Zhang



Yi Zhang

Contributors:

Qingyue Du

Yuqi Zhang

Weijian Li