NEURAL RE-RANKING FOR MULTI-STAGE RECOMMENDER SYSTEMS

RecSys Tutorial

Weiwen Liu[1], Jiarui Qin[2], Bo Chen[1], Ruiming Tang[1]

[1]Huawei Noah's Ark Lab
[2]Shanghai Jiao Tong University

Presenter Bio

Weiwen Liu

Researcher

Huawei Noah's Ark Lab

Jiarui Qin

Ph.D. student

Shanghai Jiao Tong University

Bo Chen

Researcher

Huawei Noah's Ark Lab

Ruiming Tang

Lab director

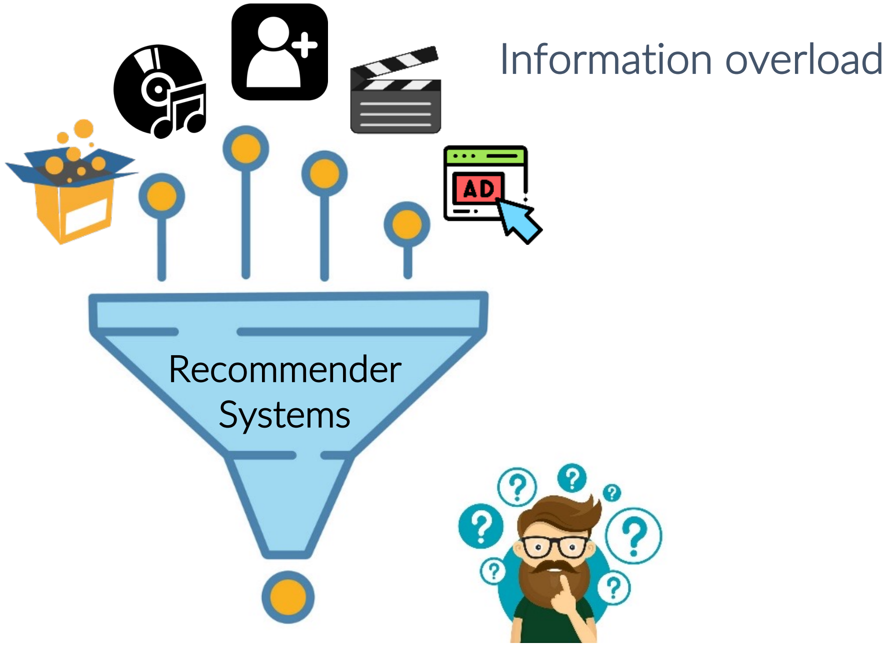
Huawei Noah's Ark Lab

Outline

- Introduction
  - Multi-stage recommender systems
  - Neural re-ranking
- Single objective: Accuracy oriented
  - Learning by observed signals
  - Learning by counterfactual signals
  - LibRerank library
- Multi-objective
  - Diversity-aware re-ranking
  - Fairness-aware re-ranking
- Emerging applications
- Summary

Everything is Recommendation

Age of Information Explosion



Recommend item X to user

Items can be Products, News, Movies, Videos, Friends, etc.

Everything is Recommendation

- Recommendation has been widely applied in online services:
  - **E-commerce**, content sharing, social networking …

amazon

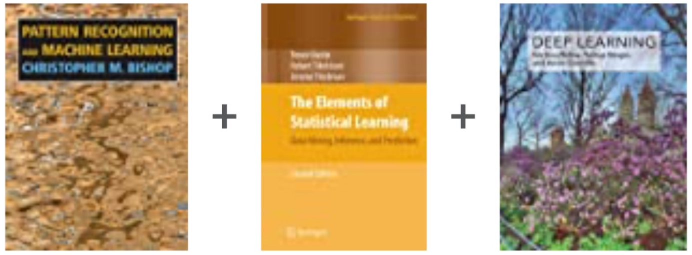
ebay

淘宝网
Taobao.com

JD.COM

Product recommendation

Frequently bought together



A B C

Total price: $208.9

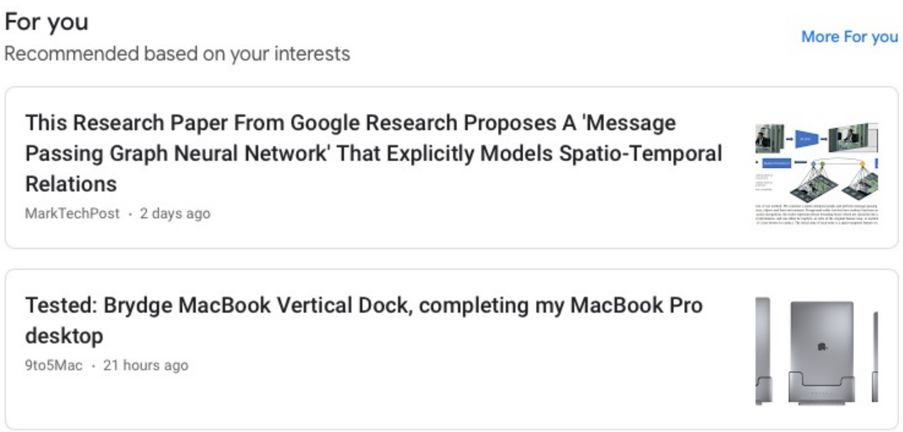
Add all three to Cart

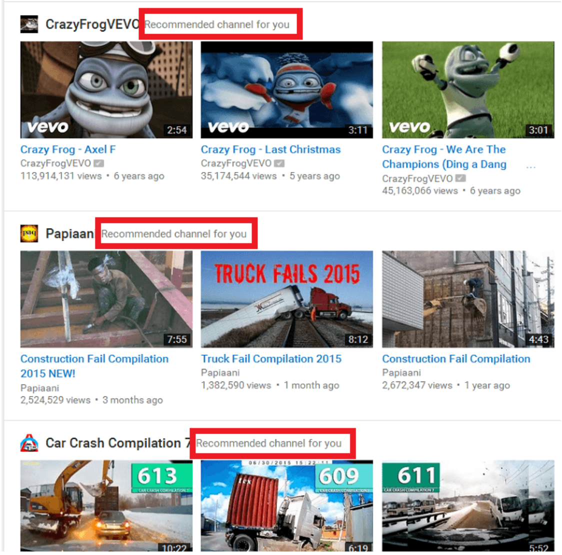
Add all three to List

Everything is Recommendation

- Recommendation has been widely applied in online services:
  - E-commerce, **content sharing**, social networking …

News/video/image recommendation

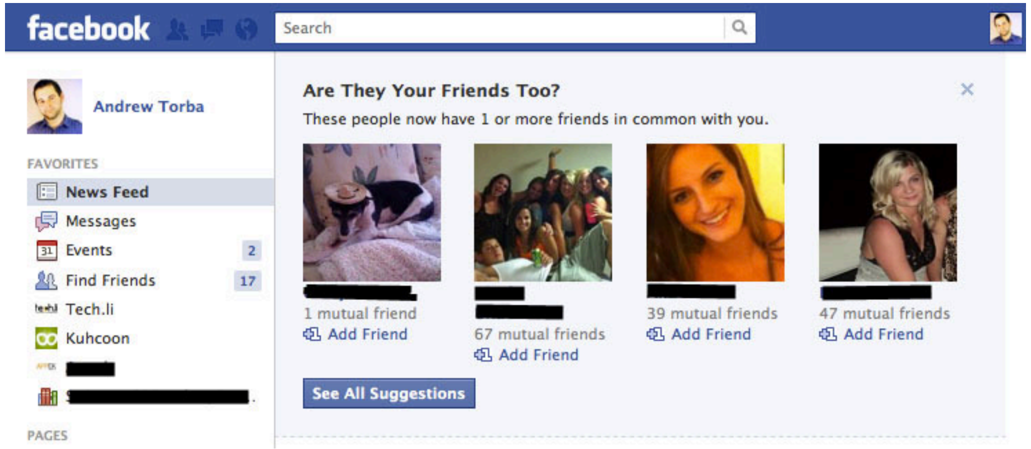




Everything is Recommendation

- Recommendation has been widely applied in online services:
  - E-commerce, content sharing, **social networking** ...

Friend recommendation



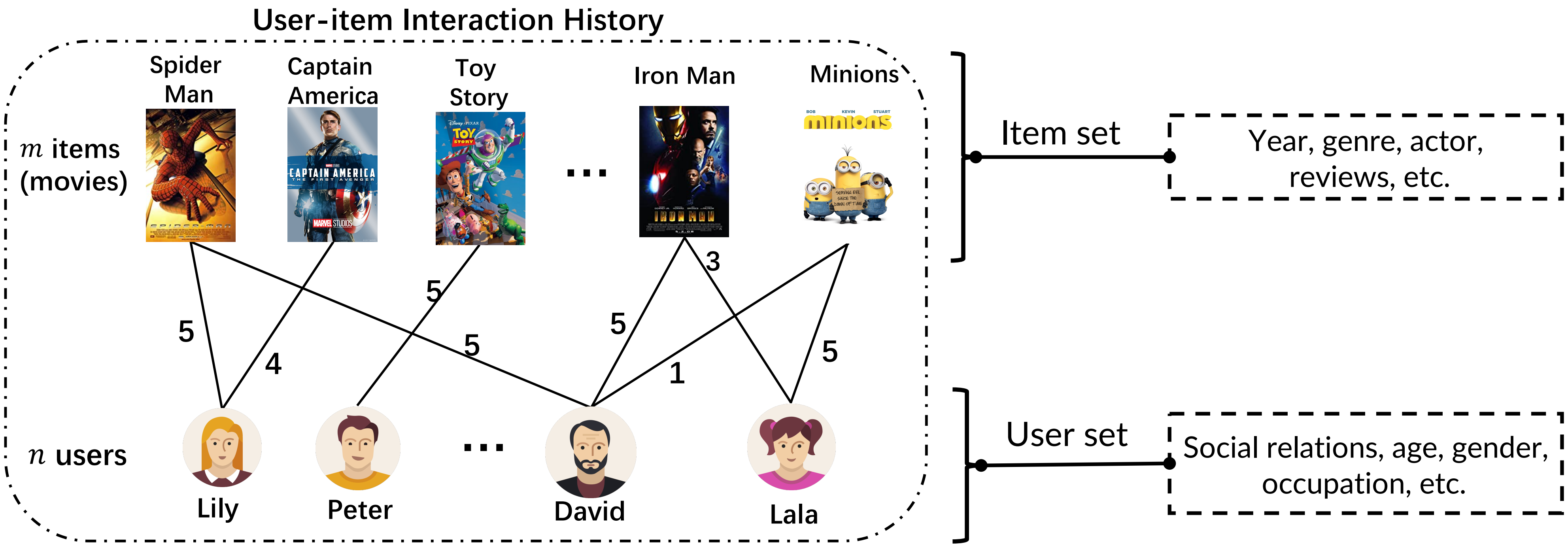
Industrial Recommender Systems

INPUT

Historical **user-item interactions** or additional **side information** (e.g., social relations, item's knowledge, etc.)

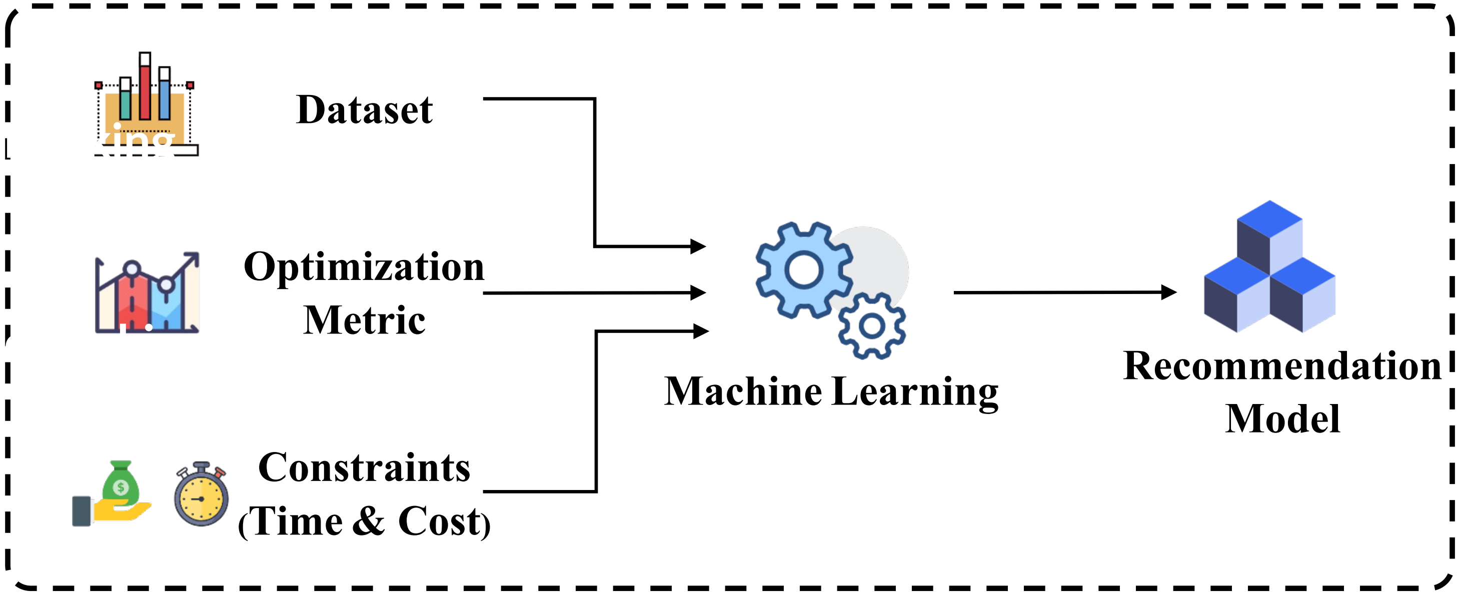
OUTPUT

Predict how likely a user would interact with a target item (e.g., click, view, or purchase)



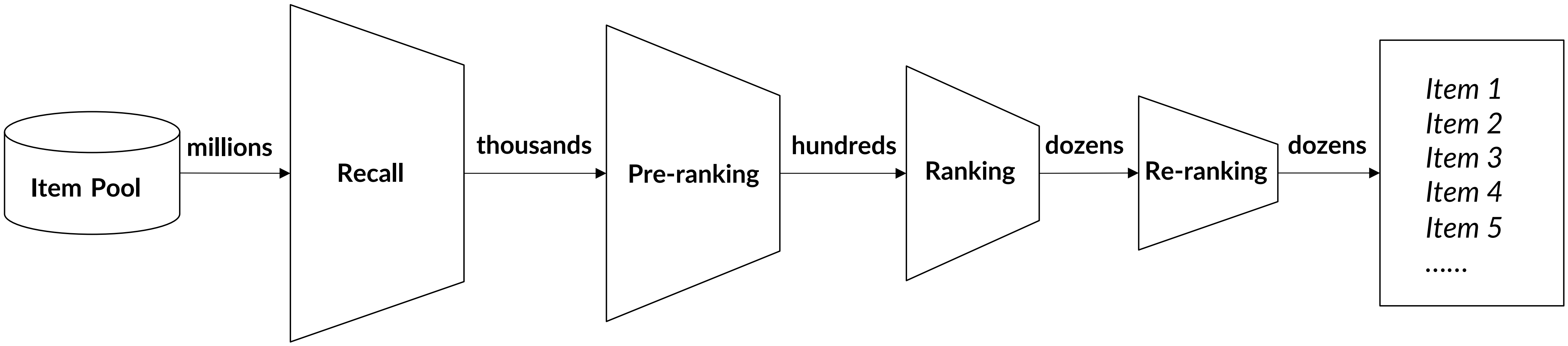
Industrial Recommender Systems

- The success of a RS algorithm is **NOT** limited to the accuracy/ranking quality
  - Business metric: eCPM, CTR, LTV, PV, VV…
  - Resource limitation: computing and memory resource.
  - Data engineering
  - Response latency



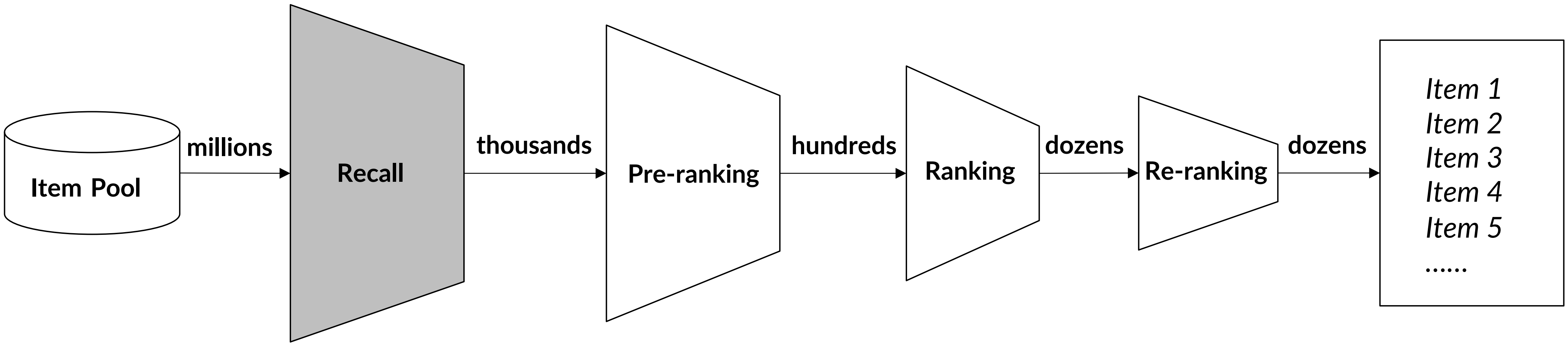
Multi-stage Recommender Systems

- The recommendation task is split into multiple stages
  - Each stage narrows down the relevant items
  - To balance the **effectiveness-efficiency trade-off**
    - Complex models are more accurate but time consuming
    - Simple models are less accurate but more efficient



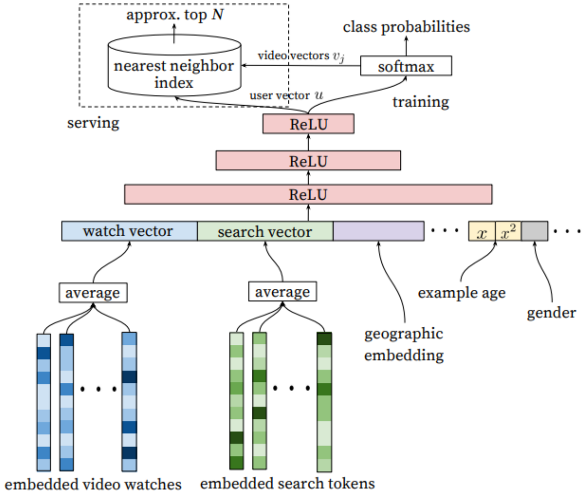
Recall Stage

- Retrieve relevant items from candidate item pool **quickly**
- **Multiple retrieve strategies**
  - Rule-based (popular, category, etc.)
  - Model-based
- **Multiple objectives** (relevance, diversity, etc.)

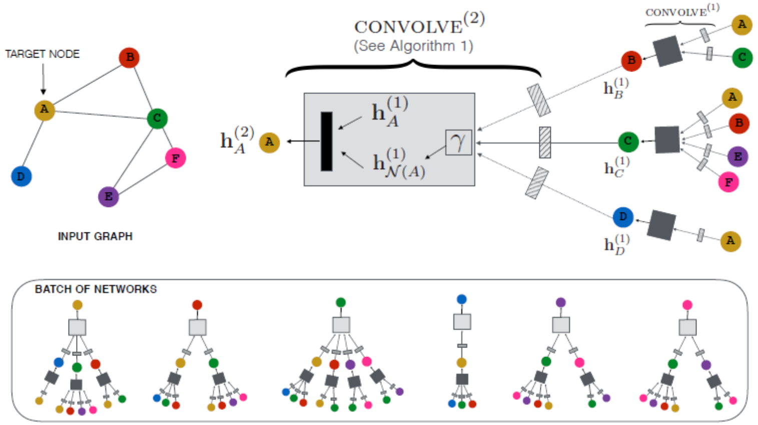


Recall Stage

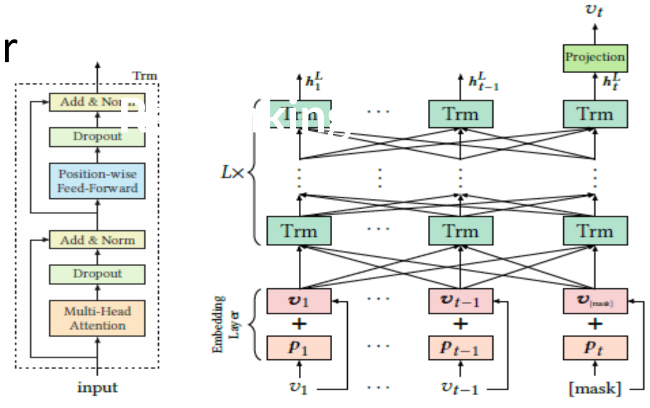
Two-tower



Graph

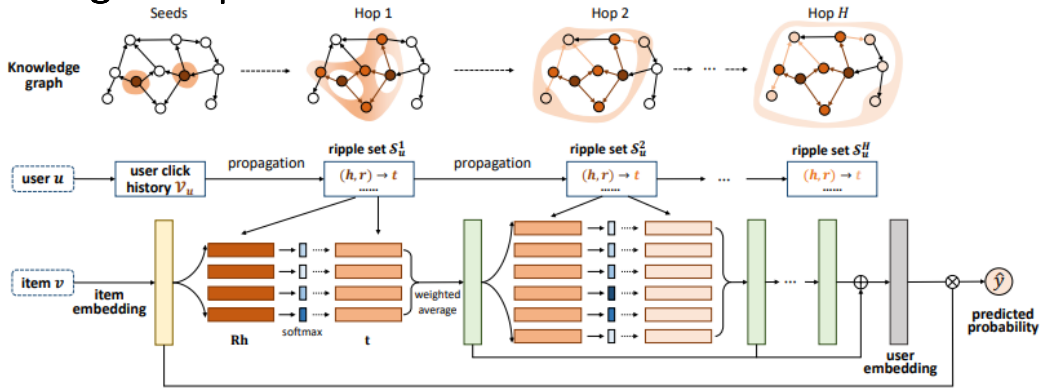


Transformer



(a) Transformer Layer. (b) BERT4Rec model architecture.

Knowledge Graph



Covington, et al. “Deep Neural Networks for YouTube Recommendations.” In RecSys, 2016.

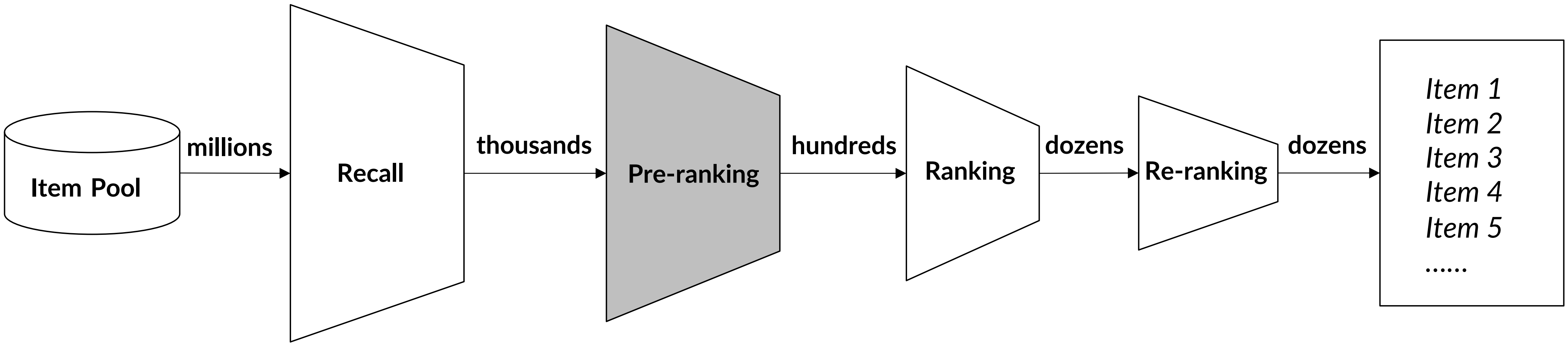
Sun, et al. “BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer.” In CIKM, 2019.

Ying, et al. "Graph Convolutional Neural Networks for Web-Scale Recommender Systems." In KDD, 2019.

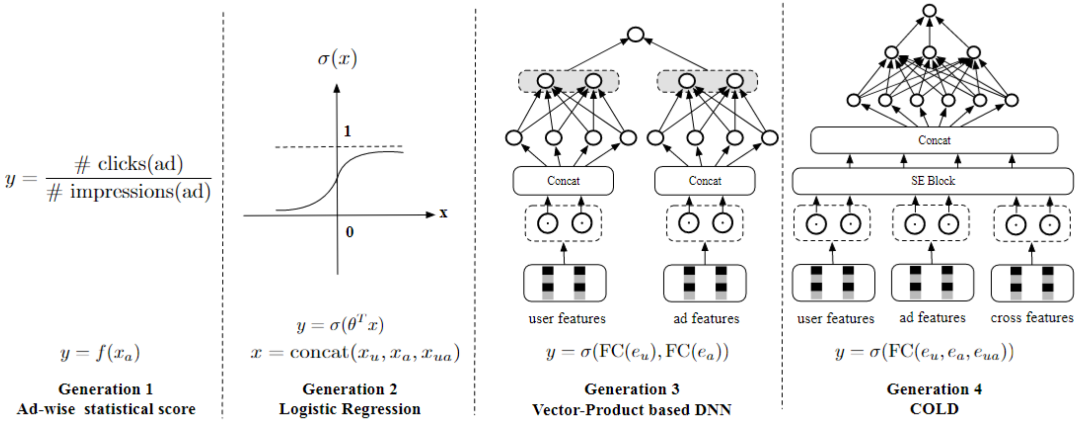
Wangle, et al. “RippleNet: Propagating User Preferences on the Knowledge Graph for Recommender Systems.” In WWW, 2018.

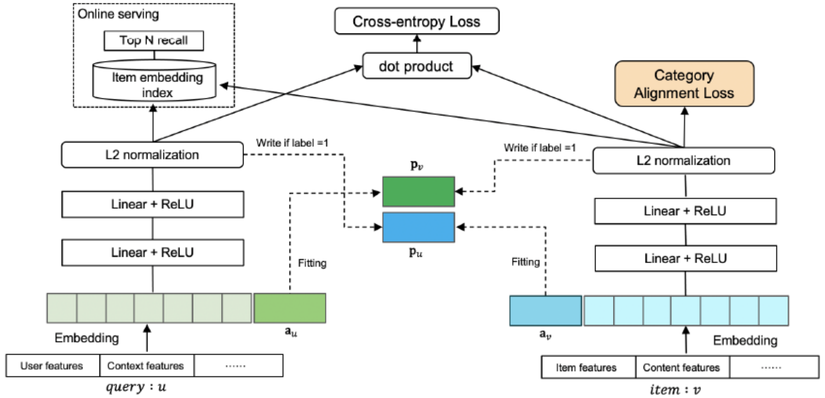
Pre-ranking Stage

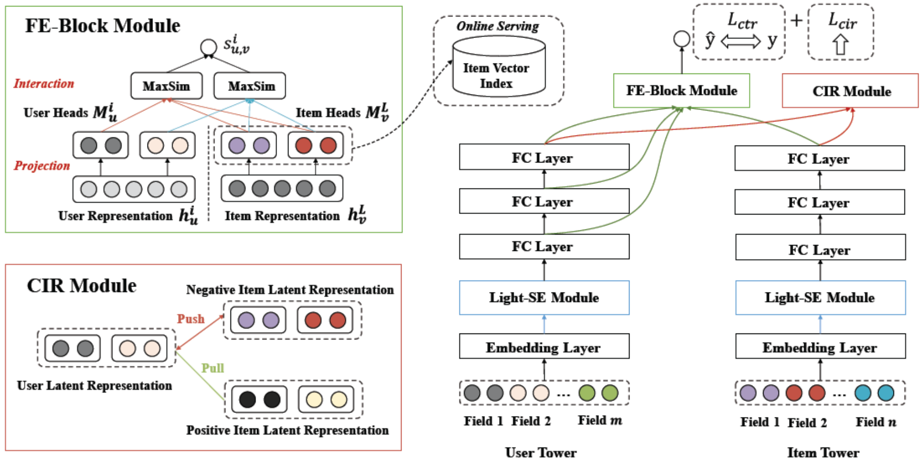
- Filter irrelevant items **efficiently**
- Balance between **efficiency and effectiveness**
- Compared to the ranking stage
  - Fewer features
  - Simpler models



Pre-ranking Stage



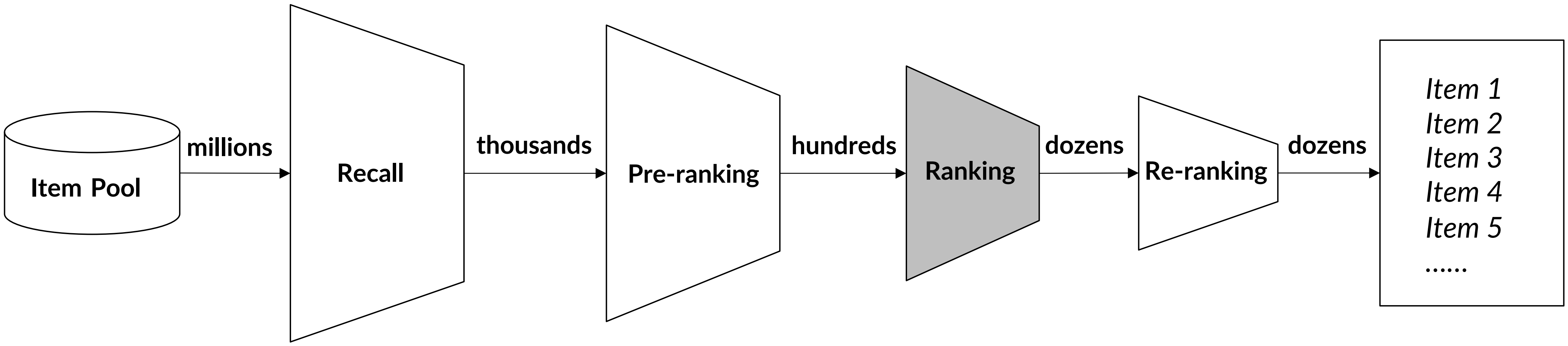




Wang Z, Zhao L, Jiang B, et al. “Cold: Towards the next generation of pre-ranking system”. arXiv preprint arXiv:2007.16122, 2020.
Yu Y, Wang W, Feng Z, et al. “A dual augmented two-tower model for online large-scale recommendation”. In DLP-KDD, 2021.
Xiangyang Li, Bo Chen, et al. “IntTower: the Next Generation of Two-Tower Model for Pre-Ranking System”, In CIKM, 2022.

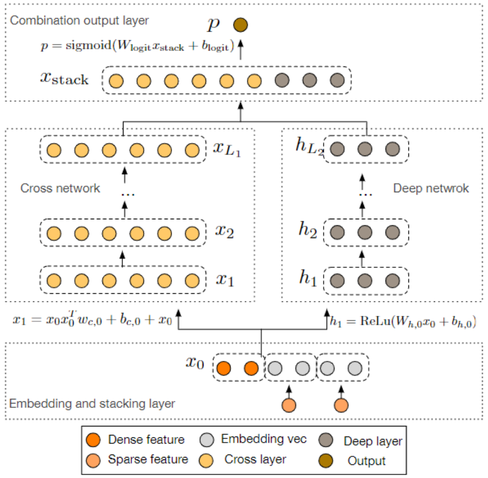
Ranking Stage

- Infer user's preference over candidate items **accurately**
- Compared to the pre-ranking stage
  - More features (including multi-modal features)
  - More complex models
    - Feature interaction, user behavior modeling
    - Multi-task, multi-domain

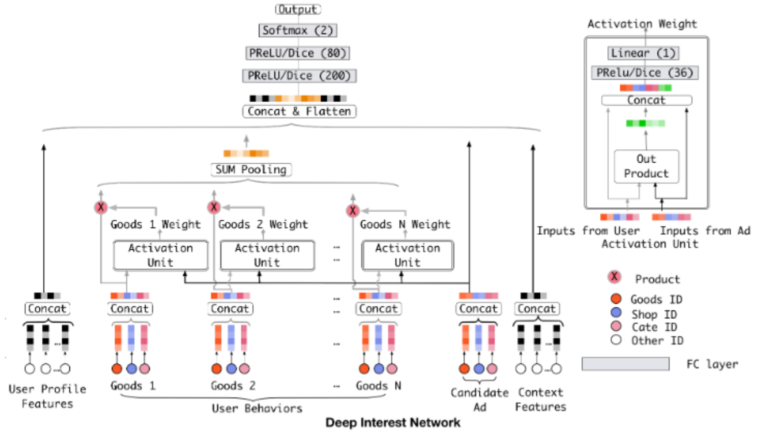


Ranking Stage

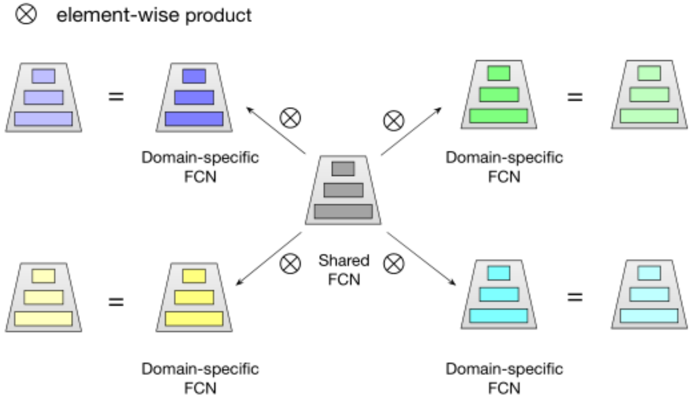
Feature interaction



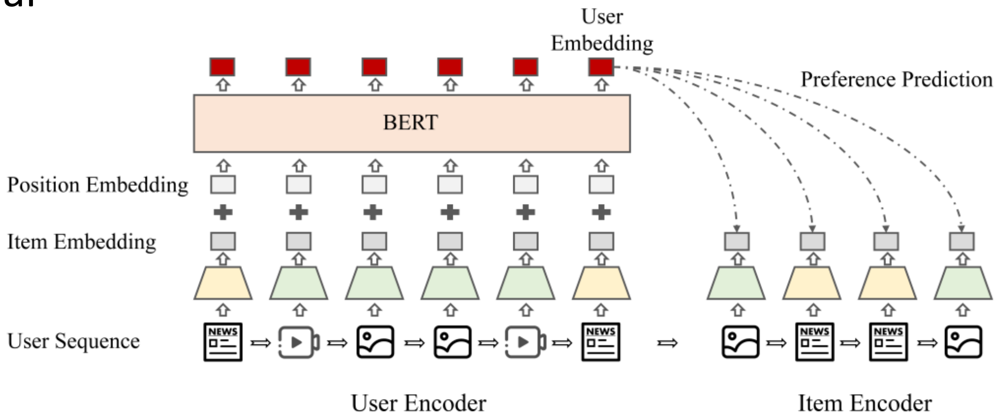
User behavior modeling



Multi-task



Multi-modal



Wang R, Fu B, Fu G, et al. "Deep & cross network for ad click predictions". In ADKDD, 2017

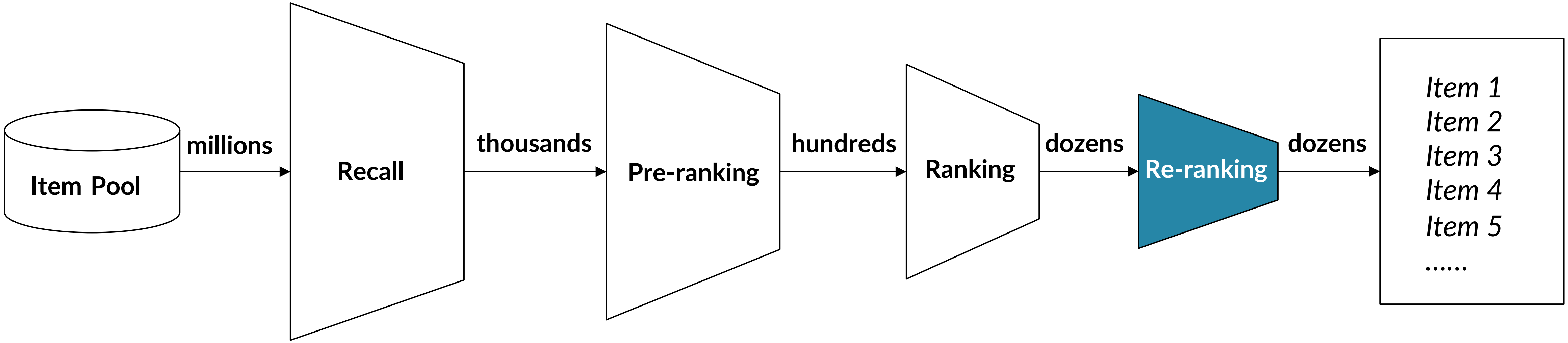
Zhou G, Zhu X, Song C, et al." Deep interest network for click-through rate prediction." In KDD 2018

Sheng, Xiang-Rong, et al. "One model to serve all: Star topology adaptive recommender for multi-domain ctr prediction." In CIKM, 2021.

Li, Yuan, et al. "TransRec: Learning Transferable Recommendation from Mixture-of-Modality Feedback". arXiv preprint arXiv:2206.06190, 2022.

Re-ranking Stage

- Re-arrange the input ranking list according to the objectives
- Consider **listwise-context/cross-item interaction**
- **Multiple objectives** (relevance, diversity, etc.)



Re-ranking Stage

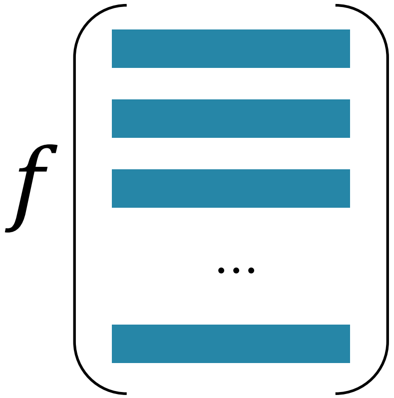
- **Formulation:**

$$\phi_* = \text{argmin}_\phi \sum_{R,Y} \mathcal{L}(Y, \phi(R))$$

where R is the input initial list contains n items, $Y \in \mathbb{R}^n$ is the supervision signal, $\mathcal{L}(\cdot)$ is the loss function, ϕ is the ranking function.

Re-ranking vs. Ranking

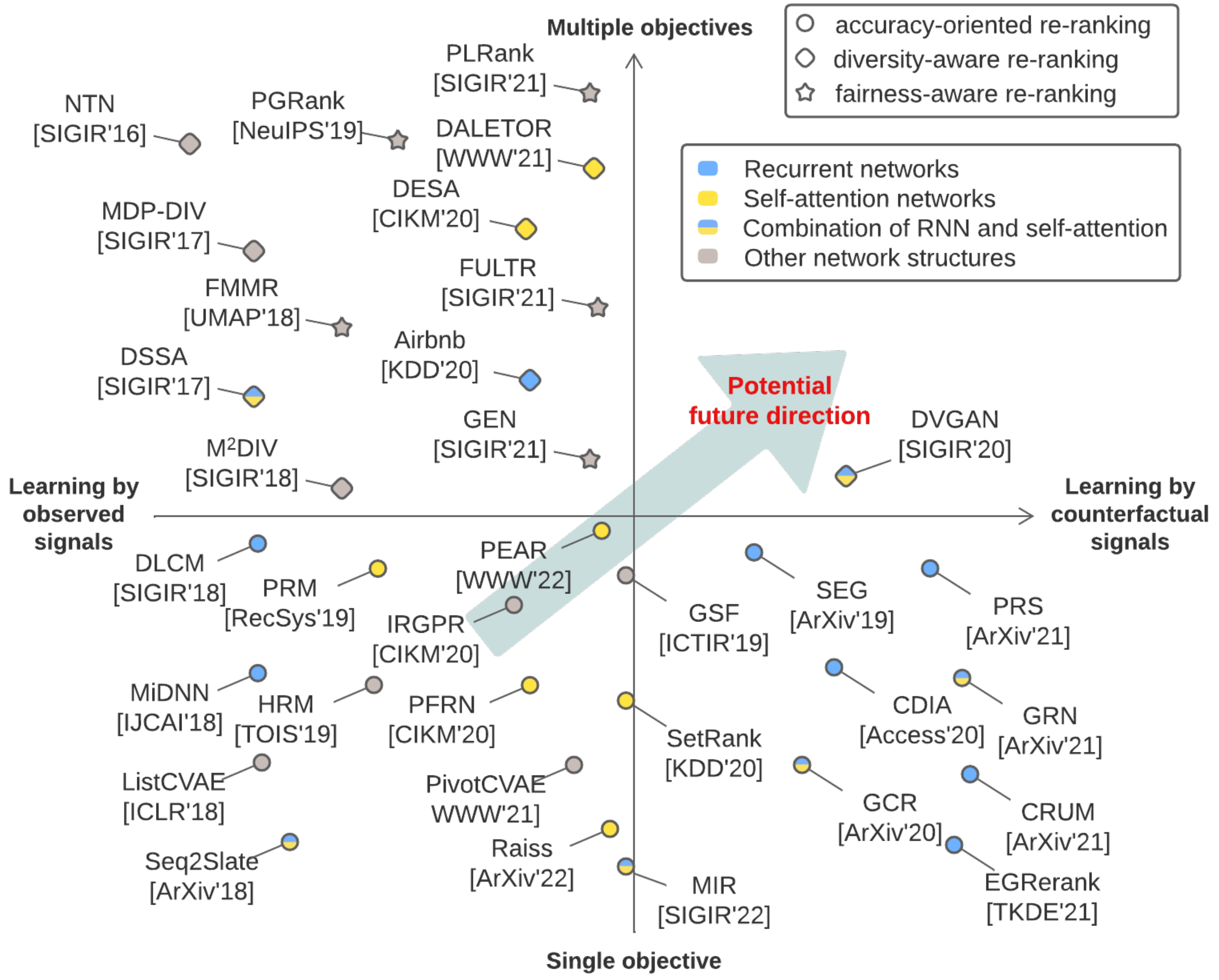
- Re-ranking
  - Multivariate
  - Take a list of items at a time
  - Cross-item interactions/mutual influences between items
  - Finding the best permutation is **NP-hard**



- Ranking
  - Univariate
  - Take one item at a time
  - Feature-level interactions within each item

A Taxonomy

- **Objectives:**
  - Accuracy-oriented re-ranking
  - Diversity-aware re-ranking
  - Fairness-aware re-ranking
- **Supervision signals:**
  - Learning by observed signals
  ⇒ **actual displayed list**
  - Learning by counterfactual signals
  ⇒ **counterfactual permutations**

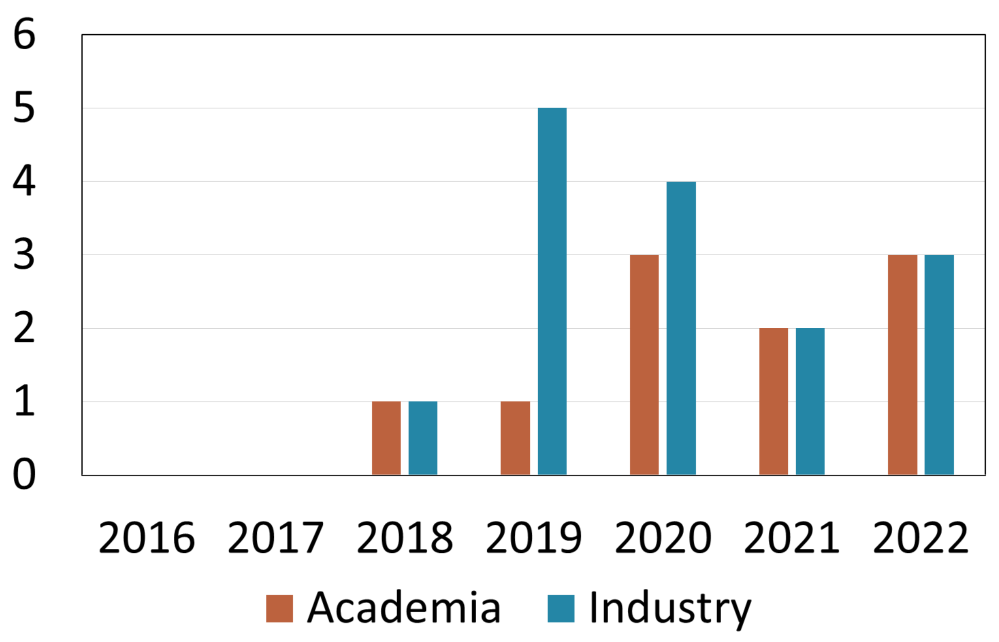


Development characteristics

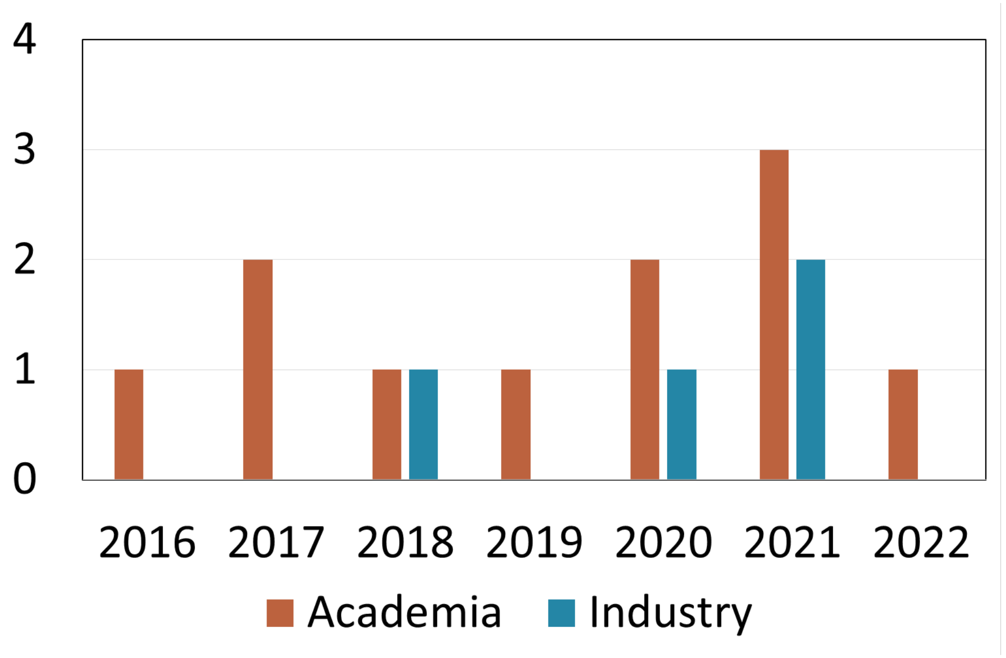
- Mostly focus on accuracy-oriented re-ranking
- Attention-based network structure becomes popular
- Few works discuss counterfactual signals for multi-objective re-ranking

Some Statistics

- The study of neural re-ranking starts in **2016**
- Till August 2022, there are about **40** papers on neural re-ranking
- **Industry** focuses more on accuracy-oriented re-ranking
- **Academia** focuses more on fairness and diversity for re-ranking



Number of papers published on accuracy-oriented re-ranking



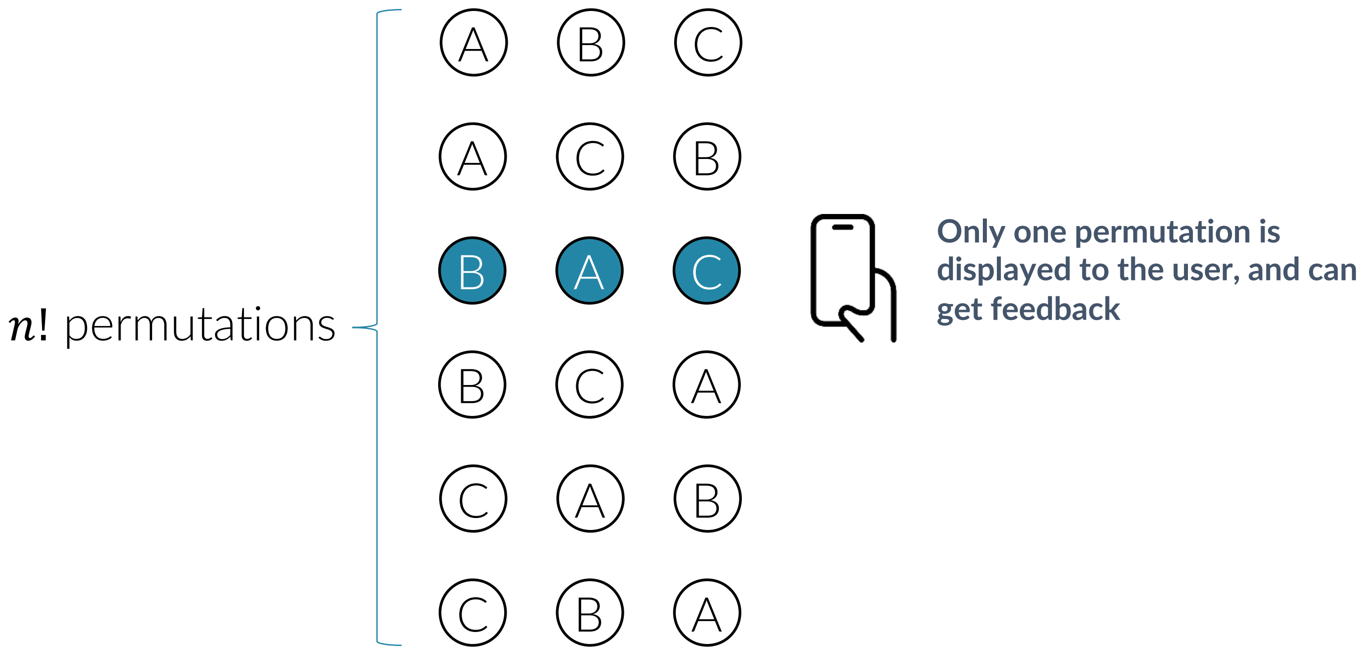
Number of papers published on diversity- or fairness-aware re-ranking

Outline

- Introduction
    - Multi-stage recommender systems
    - Neural re-ranking
- Single objective: Accuracy oriented
    - Learning by observed signals
    - Learning by counterfactual signals
    - LibRerank library
- Multi-objective
    - Diversity-aware re-ranking
    - Fairness-aware re-ranking
- Emerging applications
- Summary

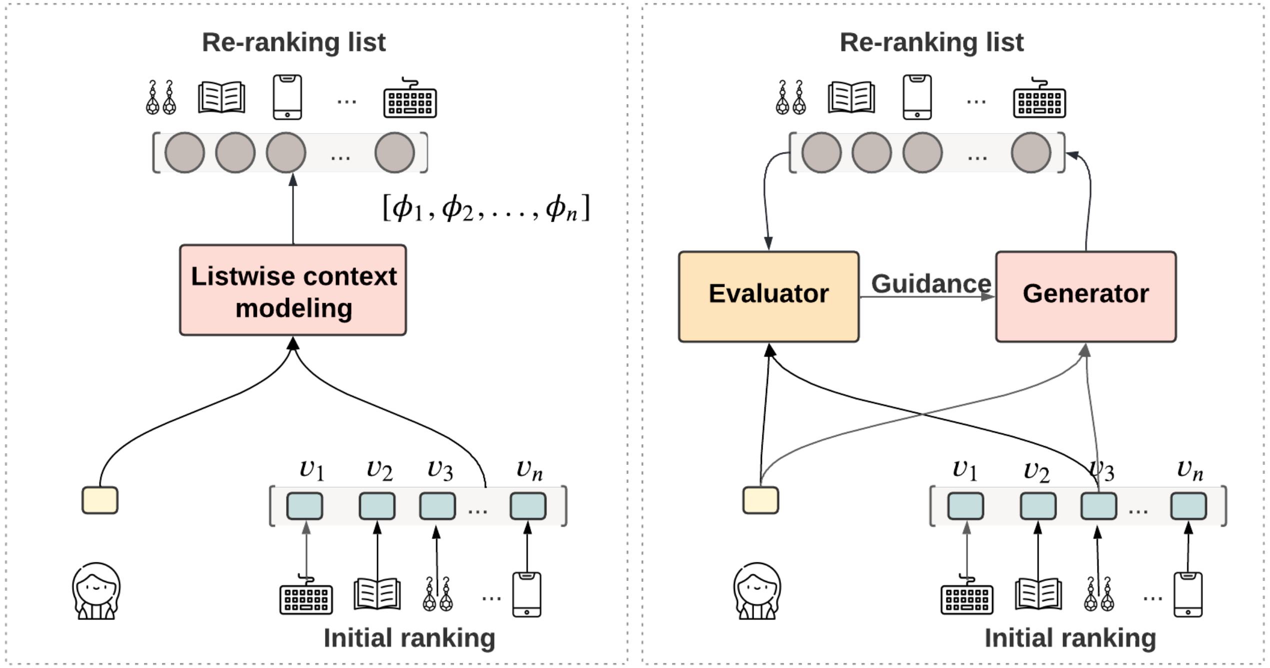
Accuracy-oriented Re-ranking

- Accuracy is the **fundamental** goal for recommender systems.
- Learning by observed signals
  - Trained by **actual displayed lists** and corresponding feedback provided by **users**
- Learning by counterfactual signals
  - Item's relevance varies under different permutations
  - Trained by **counterfactual permutations** and feedback provided by **an additional evaluator**
  - **Conterfactual lists:** permutations that have not been actually displayed to the users



Accuracy-oriented: **Network Structure**

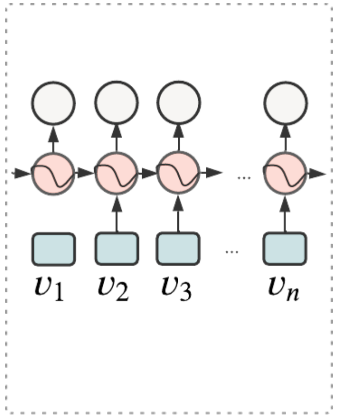
- Learning by observed signals
  - Embed user and item features into low-dimensional dense vectors
  - Extract the cross-item interactions by the listwise context modeling module
- Learning by counterfactual signals
  - Generator: generate feasible permutations
  - Evaluator: evaluate the listwise utility of each permutation



(a) (b)

Accuracy-oriented: **Learning by Observed Signals**

- Simple and straightforward.
- The actual feedback provided by users is less noisy and easier to train.
- Listwise context modeling
  - **Recurrent listwise modeling:** LSTM, GRU, BiLSTM…
  - **Attentive listwise modeling:** self-attention, cross-attention…
  - **Others:** GNN, MLP….

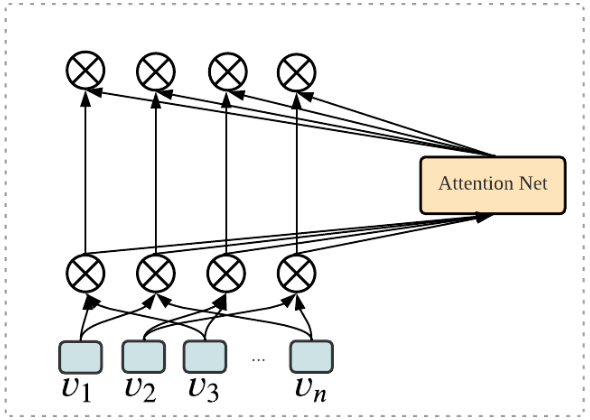


Recurrent listwise modeling

DLCM [Ai et al., 2018]

MiDNN [Zhuang et al., 2018]

Seq2Slate [Belllo et al., 2018]



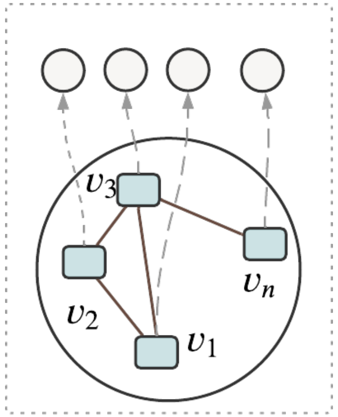
Attentive listwise modeling

PRM [Pei et al., 2019]

PFRN [Huang et al., 2020]

Raiss [Lin et al., 2022]

PEAR [Li et al., 2022]



Other network structures

List-CVAE [Jiang et al., 2019]

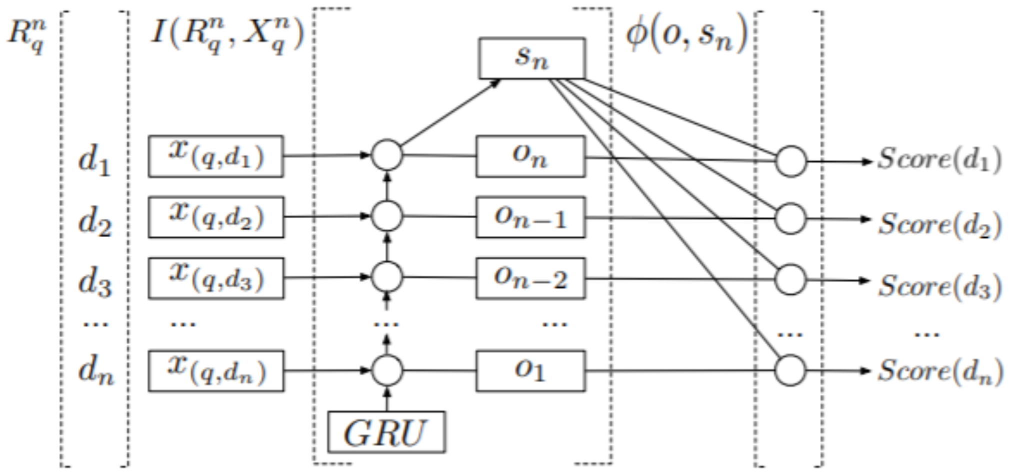
PivotCVAE [Liu et al., 2021]

HRM [Li et al., 2019]

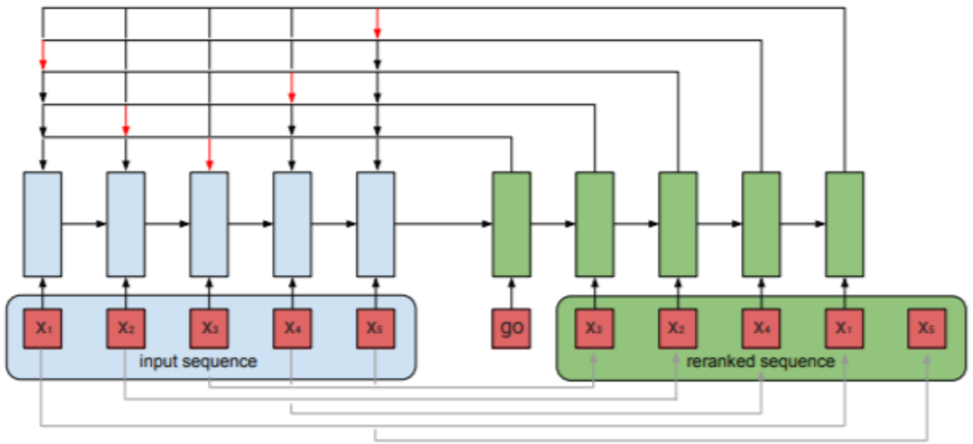
IRGPR [Liu et al., 2020]

Learning by Observed Signals: **Recurrent Listwise Modeling**

- DLCM
  - GRU
  - Capture the local ranking context of top items.



- Seq2Slate
  - Pointer network
  - At each step, predict the next "best" item.



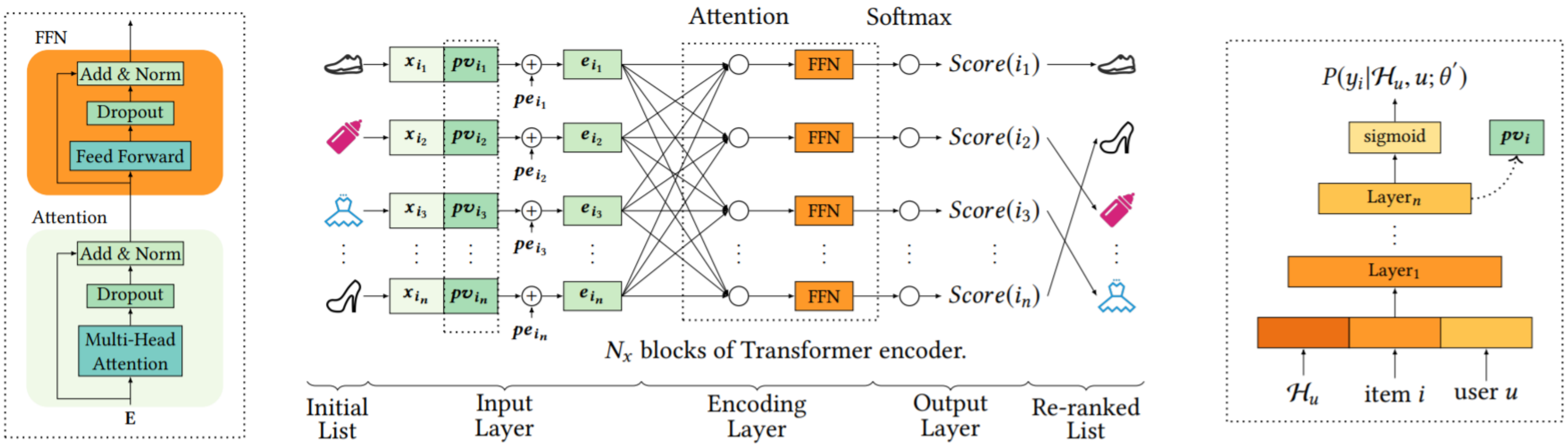
Irwan Bello, Sayali Kulkarni, Sagar Jain, et al. Seq2slate: Re-ranking and slate optimization with rnns, 2018.
Qingyao Ai, Keping Bi, Jiafeng Guo, et al. Learning a deep listwise context model for ranking refinement. In SIGIR, 2018.

Learning by Observed Signals: **Attentive Listwise Modeling**

- Personalized Re-ranking for Recommendation (PRM)
  - Self-attention mechanism
    - The influence degrades along with the encoding distance in RNN-based approaches
    - Captures mutual influences between any pair of items
    - More efficient, can be made parallel
  - Personalized re-ranking
    - Pretrained user embedding

Changhua Pei, Wenwu Ou, et al. Personalized re-ranking for recommendation. In RecSys, 2019.

Personalized Re-ranking for Recommendation



(a) One block of Transformer encoder.

(b) Architecture of PRM.

(c) The pre-trained model to generate $\boldsymbol{pv_i}$, $i = i_1, \ldots, i_n$.

Changhua Pei, Wenwu Ou, et al. Personalized re-ranking for recommendation. In RecSys, 2019.

Personalized Re-ranking for Recommendation

- Performance on public dataset: Yahoo!

Init. List	Reranking	Yahoo Letor dataset.				
		Precision@5(%)	Precision@10(%)	MAP@5(%)	MAP@10(%)	MAP(%)
SVMRank	SVMRank	50.42	42.25	73.71	68.28	62.14
	LambdaMART	51.35	43.08	74.94	69.54	63.38
	DLCM	52.54	43.26	76.52	70.86	64.50
	PRM-BASE	**53.29**	**43.66**	**77.62**	**72.02**	**65.60**
LambdaMART	SVMRank	50.41	42.34	73.82	68.27	62.13
	LambdaMART	52.04	43.00	75.77	70.49	64.04
	DLCM	52.54	43.16	77.81	71.88	65.24
	PRM-BASE	**53.63**	**43.41**	**78.62**	**72.67**	**65.72**

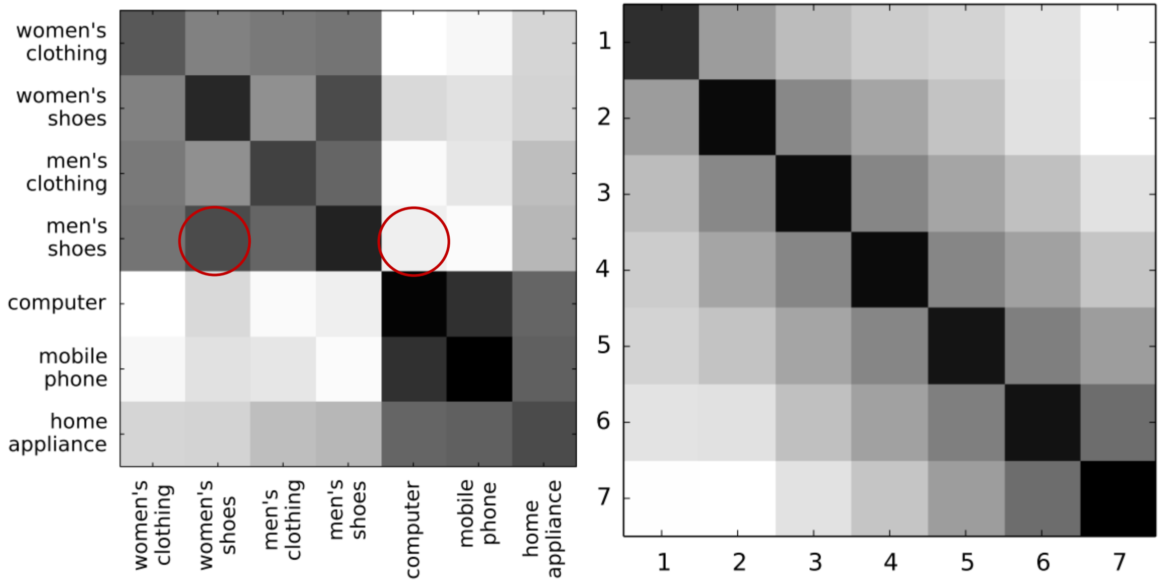
- Online A/B test on an e-commerce platform

Reranking	PV	IPV	CTR	GMV
DLCM	0.77%	1.75%	0.97%	0.13%
PRM-BASE	1.27%	2.44%	1.16%	0.36%
PRM-Personalized-Pretrain	**3.01%**	**5.69%**	**2.6%**	**6.65%**

Changhua Pei, Wenwu Ou, et al. Personalized re-ranking for recommendation. In RecSys, 2019.

Personalized Re-ranking for Recommendation

- Visualization of the attention weights

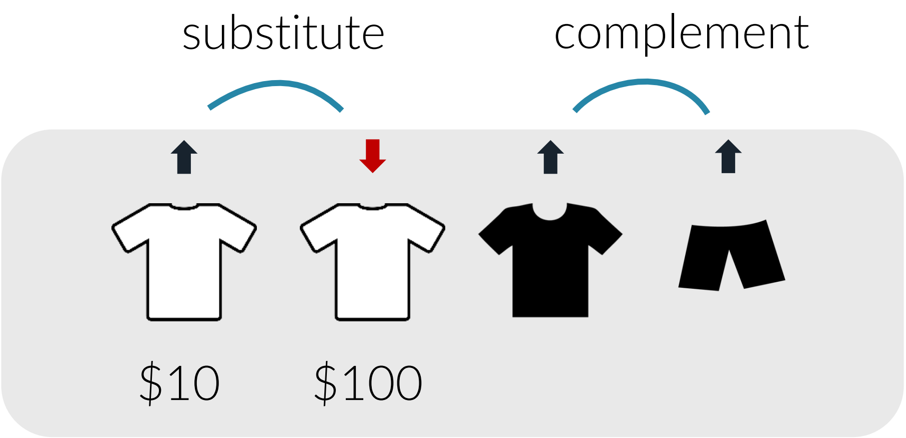


(a) Category. (b) Price.

Changhua Pei, Wenwu Ou, et al. Personalized re-ranking for recommendation. In RecSys, 2019.

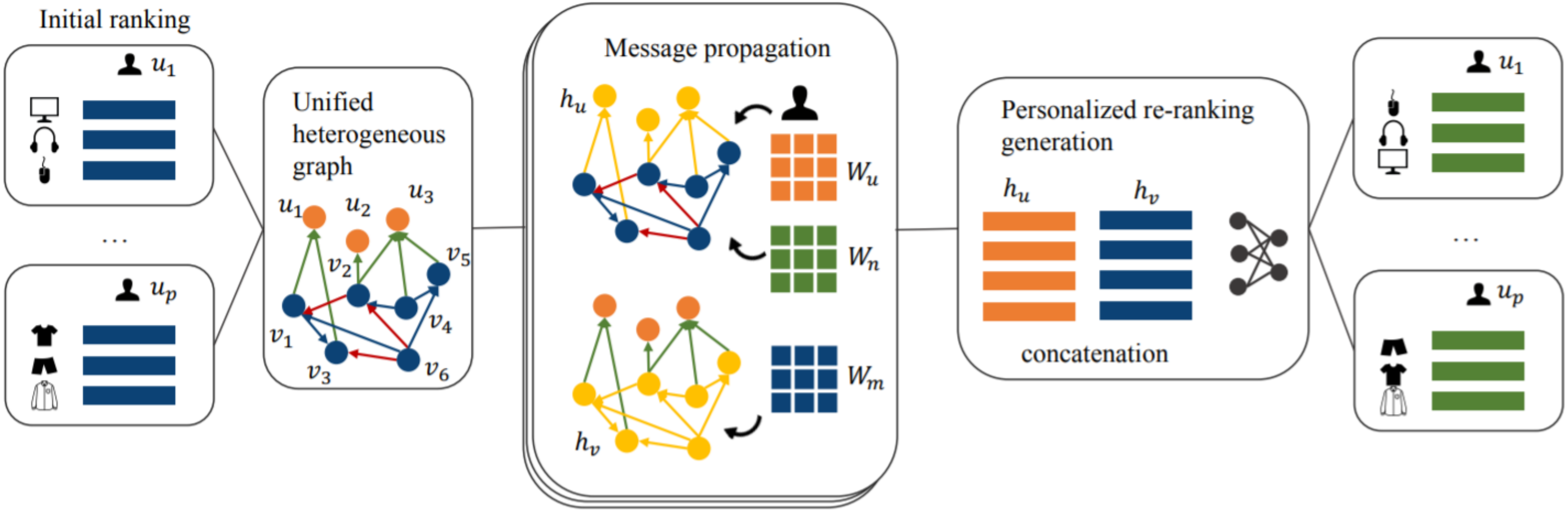
Learning by Observed Signals: **Graph Representation Learning**

- Personalized re-ranking with item relationships for e-commerce (IRGPR)
  - Item relationships affect the behavior of a user on this list
    - **Substitutable:** items are interchangeable, co-click
    - **Complementary:** items are bought together by users
  - Personalized user preferences and intents
    - Prices, quality…



Weiwen Liu, Qing Liu, Ruiming Tang, Junyang Chen, Xiuqiang He, and Pheng Heng. Personalized re-ranking with item relationships for e-commerce. In CIKM, 2020.

Personalized Re-ranking with Item Relationships for E-commerce



Weiwen Liu, Qing Liu, Ruiming Tang, Junyang Chen, Xiuqiang He, and Pheng Heng. Personalized re-ranking with item relationships for e-commerce. In CIKM, 2020.

Personalized Re-ranking with Item Relationships for E-commerce

- **Message propagation steps:**

  Learn item/user representation vectors (h_v/h_u).

$$h_v^{(l+1)} = \frac{1}{|\mathcal{N}_v|} \sum_{w \in \mathcal{N}_v} A h_w^{(l)} .$$

Initial ranking scores

User node $\boldsymbol{u}$:

$A = MLP(x_u) \cdot MLP(e_{vu})$

Item node $\boldsymbol{v}$:

$A = MLP(e_{vw})$

Intent embedding network

item relationships

- **Personalized re-ranking:**

$$\hat{y}_{uv} = \sigma\left(MLP\left(h_u^{(L)} \middle|\middle| h_v^{(L)}\right)\right).$$

Weiwen Liu, Qing Liu, Ruiming Tang, Junyang Chen, Xiuqiang He, and Pheng Heng. Personalized re-ranking with item relationships for e-commerce. In CIKM, 2020.

Personalized Re-ranking with Item Relationships for E-commerce

- **Amazon Dataset [McAuley'15]**
  - **Also Bought (AB):** Users bought x also bought y across sessions;
  - **Also Viewed (AV):** Users viewed x also viewed y;
  - **Bought Together (BT):** Users frequently bought x and y (x and y were purchased as part of a single basket);
  - **Buy after Viewing (BV):** Users who viewed x eventually bought y.

categories	#user	#item	#rating	density (1e-3)	#AB	#AV	#BT	#BV
Video Games	2,390	48,938	148,420	1.27	1,143,763	170,107	27,460	117,400
Musical Instruments	565	65,150	31,806	0.86	531,379	480,710	26,955	117,902
Movies & TV	18,193	200,515	1,800,336	0.49	2,766,430	172,940	80,924	224,627
Electronics	16,187	424,116	847,556	0.12	2,550,227	2,823,653	126,166	769,868
Clothing, Shoes, and Jewelry	9,746	755,510	463,774	0.06	2,188,897	5,875,987	208,744	1,693

Weiwen Liu, Qing Liu, Ruiming Tang, Junyang Chen, Xiuqiang He, and Pheng Heng. Personalized re-ranking with item relationships for e-commerce. In CIKM, 2020.

Personalized Re-ranking with Item Relationships for E-commerce

- Performance on public dataset：Amazon

		Precision@5	MAP@5	Precision@10	MAP@10	Precision@20	MAP@20
Video Games	DeepFM	0.7506	0.8137	0.7494	0.7983	0.6967	0.7774
	DLCM	0.7554	0.8238	0.7476	0.8047	0.6923	0.7812
	PRM	0.7651	0.8310	0.7561	0.8081	0.6795	0.7842
	IRGPR	**0.8241***	**0.8956***	**0.7855***	**0.8584***	**0.7019***	**0.8169**
	imp.%	+7.71%	+7.77%	+3.89%	+6.22%	+0.75%	+4.17%
Musical Instruments	DeepFM	0.6056	0.7405	0.5111	0.6893	0.5089	0.5984
	DLCM	0.6111	0.7517	0.5528	0.6957	0.4931	0.6201
	PRM	0.6233	0.7750	0.5472	0.7152	0.5028	0.6169
	IRGPR	**0.6751***	**0.8285***	**0.5573***	**0.7607***	**0.5101***	**0.7364***
	imp.%	+8.31%	+6.90%	+0.81%	+6.36%	+0.24%	+18.76%
Movies & TV	DeepFM	0.7398	0.8692	0.6724	0.8102	0.6008	0.7419
	DLCM	0.7745	0.8428	0.7752	0.8239	0.6493	0.8098
	PRM	0.8077	0.8841	0.7544	0.8577	0.6596	0.8216
	IRGPR	**0.8300***	**0.8945***	**0.7862***	**0.8664***	**0.6859***	**0.8239**
	imp.%	+2.76%	+1.18%	+1.42%	+1.01%	+3.99%	+0.28%
Electronics	DeepFM	0.8068	0.9349	0.6925	0.8675	0.5947	0.7795
	DLCM	0.8266	0.8984	0.7726	0.8685	0.6311	0.7875
	PRM	0.8261	0.9185	0.7897	0.8705	**0.6490**	0.8233
	IRGPR	**0.8776***	**0.9386***	**0.8031***	**0.9026***	0.6472	**0.8481***
	imp.%	+6.17%	+0.40%	+1.70%	+3.69%	-0.28%	+3.01%
Clothing, Shoes, and Jewelry	DeepFM	0.5970	0.7859	0.5619	0.7071	0.5281	0.6427
	DLCM	0.6872	0.7915	0.6087	0.7463	0.5402	0.6761
	PRM	0.6811	0.7989	0.6358	0.7546	**0.5748**	0.7000
	IRGPR	**0.7057***	**0.8426***	**0.6381***	**0.7878***	0.5628	**0.7176***
	imp.%	+2.69%	+5.47%	+0.36%	+4.40%	-2.09%	+2.51%

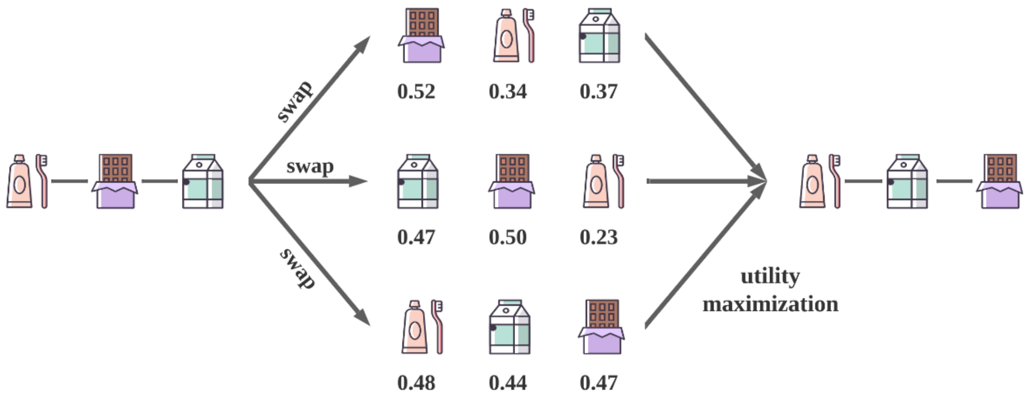
We conduct a two-sided significant test between the proposed IRGPR and the strongest baseline, where * means the p-value is smaller than 0.05. imp.% computes the improvement achieved by IRGPR over the strongest baseline.

Weiwen Liu, Qing Liu, Ruiming Tang, Junyang Chen, Xiuqiang He, and Pheng Heng. Personalized re-ranking with item relationships for e-commerce. In CIKM, 2020.

Learning by Observed Signals: **Limitation**

- Trained with the only permutation that is displayed to the users.
- Other $n! - 1$ permutation un-explored
- Early-scoring problem
  - Learning by observed signals only models the listwise context of the initial lists
  - Re-ranking operation changes the listwise context

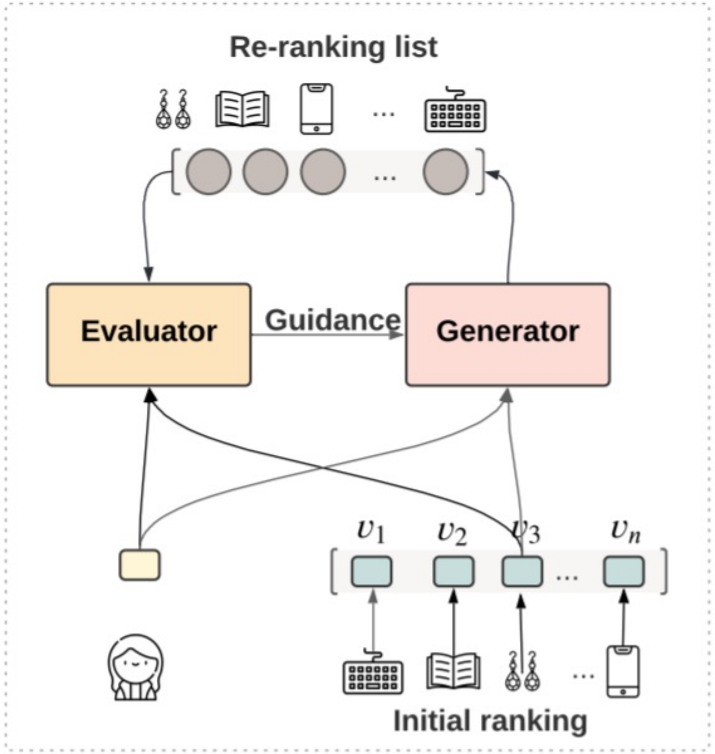
Scoring before re-ranking



Scoring after re-ranking

Accuracy-oriented: **Learning by Counterfactual Signals**

- Challenges
  - Scoring after reranking: Impractical to ask for feedback for each counterfactual list
  - Exponential time complexity: Combinatorial optimization problem, $n!$ Feasible permutations.
- Evaluator-generator paradigm
  - Evaluator: Evaluate listwise utility
  - Generator: Generate feasible permutations

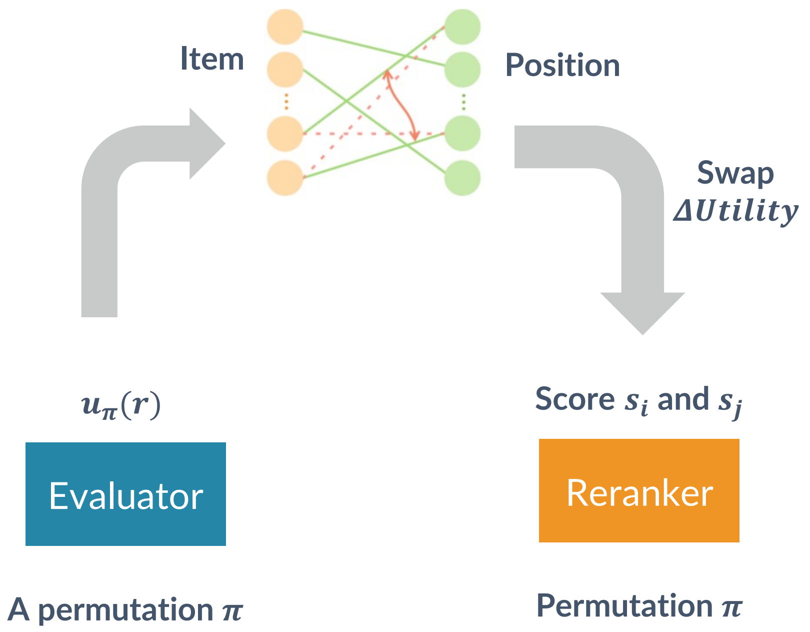


Utility-oriented Re-ranking with Counterfactual Context

- Position-aware graph embedding
  - Capture item-item and item-position correlations
- Utility-oriented evaluator
  - BiLSTM
  - Estimate the listwise utility of any permutation
- Reranker
  - MLP
- Ideal loss: $\mathcal{L}(r) = u_{\pi_*}(r) - u_{\pi}(r)$ ⇒ Undifferentiable!
- Pairwise optimization with Lambdaloss

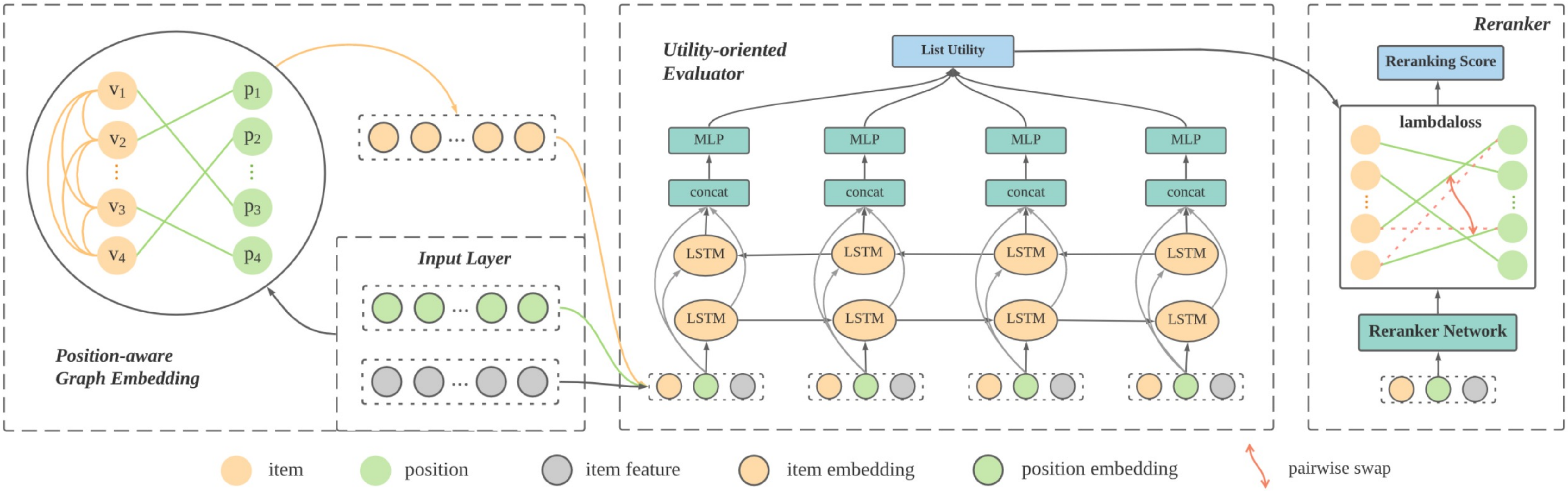
$$\mathcal{L}_{\lambda}(r;\Theta) = \sum_{i=1}^{n} \sum_{j:k_i>k_j} \Delta Utility(i,j) \log(1 + e^{-\sigma(s_i - s_j)})$$

$$\Delta Utility(i,j) = u_{\pi'}(r) - u_{\pi}(r)$$



Xi, Yunjia, Weiwen Liu, Xinyi Dai, Ruiming Tang, Weinan Zhang, Qing Liu, Xiuqiang He, and Yong Yu. "Utility-oriented re-ranking with counterfactual context." *arXiv preprint arXiv:2110.09059* (2021).

Utility-oriented Re-ranking with Counterfactual Context: **Framework**



Xi, Yunjia, Weiwen Liu, Xinyi Dai, Ruiming Tang, Weinan Zhang, Qing Liu, Xiuqiang He, and Yong Yu. "Utility-oriented re-ranking with counterfactual context." *arXiv preprint arXiv:2110.09059* (2021).

Utility-oriented Re-ranking with Counterfactual Context: **Experiments**

- Performance on two public datasets: MSLR and Yahoo!

Initial Ranker	Reranking Model	Microsoft MSLR-WEB10K					Yahoo! LETOR set 1				
		Relevance-based			Utility-based		Relevance-based			Utility-based	
		MAP	nDCG@5	nDCG@10	# Click	CTR	MAP	nDCG@5	nDCG@10	# Click	CTR
DNN	None	0.5338	0.4970	0.6775	1.7180	0.1744	0.6935	0.6704	0.8012	2.8978	0.3151
	DLCM	0.5397	0.5052	0.6818	1.7730	0.1768	0.7141	0.6958	0.8170	2.9810	0.3267
	Seq2Slate	0.5470	0.5170	0.6888	1.8046	0.1789	0.7267	0.7006	0.8167	3.0590	0.3327
	PRM	0.5531	0.5232	0.6945	1.7767	0.1808	0.7265	0.7001	0.8178	3.0642	0.3332
	SetRank	0.5404	0.5095	0.6820	1.7563	0.1770	0.7160	0.6973	0.8176	3.0111	0.3275
	URCC	**0.5827***	**0.5557***	**0.7161***	**1.8712***	**0.1860***	**0.7385***	**0.7216***	**0.8343***	**3.0915***	**0.3409***
SVMRank	None	0.5279	0.4982	0.6735	1.7156	0.1720	0.6790	0.6518	0.7914	2.7957	0.3032
	DLCM	0.5365	0.5032	0.6957	1.7816	0.1755	0.7102	0.6898	0.8143	2.9875	0.3219
	Seq2Slate	0.5494	0.5177	0.6915	1.7857	0.1788	0.7166	0.6974	0.8192	2.9791	0.3245
	PRM	0.5540	0.5224	0.6957	1.8391	0.1798	0.7187	0.7001	0.8206	2.9789	0.3252
	SetRank	0.5377	0.5059	0.6805	1.8126	0.1759	0.7090	0.6885	0.8132	2.9715	0.3214
	URCC	**0.5806***	**0.5578***	**0.7142***	**1.8599***	**0.1853***	**0.7353***	**0.7192***	**0.8328***	**3.0801***	**0.3379***
LambdaMART	None	0.5465	0.5124	0.6866	1.7726	0.1769	0.7140	0.6925	0.8188	2.9791	0.3239
	DLCM	0.5506	0.5190	0.6897	1.8003	0.1789	0.7260	0.7075	0.8267	3.0512	0.3318
	Seq2Slate	0.5686	0.5434	0.7050	1.8028	0.1826	0.7331	0.7167	0.8318	3.0570	0.3344
	PRM	0.5699	0.5438	0.7059	1.8266	0.1828	0.7339	0.7163	0.8327	3.0661	0.3351
	SetRank	0.5531	0.5235	0.6911	1.7922	0.1792	0.7282	0.7084	0.8276	3.0539	0.3329
	URCC	**0.5855***	**0.5629***	**0.7165***	**1.8688***	**0.1857***	**0.7360***	**0.7197***	**0.8335***	**3.1116***	**0.3392***

∗ denotes statistically significant improvement (measured by t-test with p-value<0.05) over all baselines.

Xi, Yunjia, Weiwen Liu, Xinyi Dai, Ruiming Tang, Weinan Zhang, Qing Liu, Xiuqiang He, and Yong Yu. "Utility-oriented re-ranking with counterfactual context." *arXiv preprint arXiv:2110.09059* (2021).

Utility-oriented Re-ranking with Counterfactual Context: **Experiments**

- Performance on real-world dataset: App Store

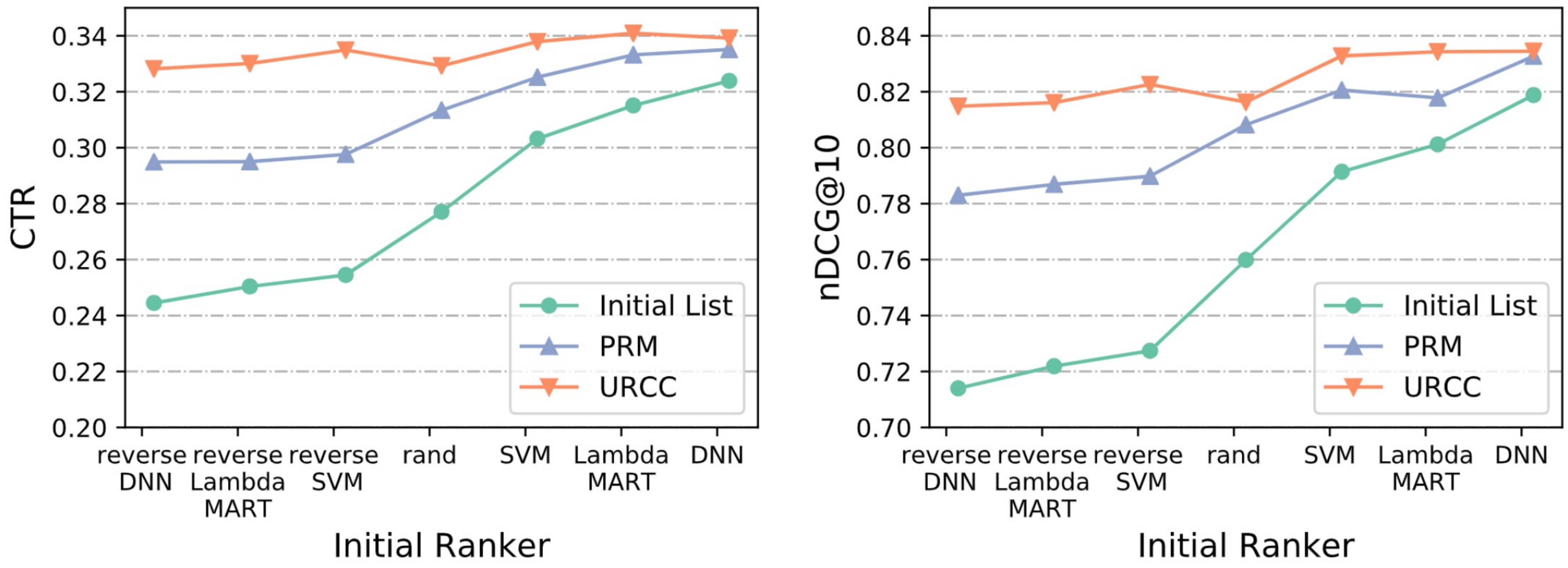
Reranking Model	nDCG@3	nDCG@5	nDCG@10	nDCG@20	Revenue@3	Revenue@5	Revenue@10	Revenue @20
initial	0.3045	0.3646	0.4281	0.4848	2.461	3.313	4.292	5.304
DLCM	0.2943	0.3631	0.4323	0.4839	2.634	3.478	4.461	5.393
Seq2Slate	0.2861	0.3541	0.4223	0.4776	2.640	3.468	4.466	5.390
PRM	0.3020	0.3648	0.4342	0.4883	2.676	3.478	4.484	5.401
SetRank	0.3077	0.3842	0.4599	0.4993	2.491	3.380	4.421	5.353
URCC	**0.3477***	**0.4127***	**0.4802***	**0.5255***	**2.688***	**3.503***	**4.494***	**5.405**

∗ denotes statistically significant improvement (measured by t-test with p-value<0.05) over all baselines.

Xi, Yunjia, Weiwen Liu, Xinyi Dai, Ruiming Tang, Weinan Zhang, Qing Liu, Xiuqiang He, and Yong Yu. "Utility-oriented re-ranking with counterfactual context." *arXiv preprint arXiv:2110.09059* (2021).

Utility-oriented Re-ranking with Counterfactual Context: **Experiments**

- Sensitivity to the performance of the initial lists



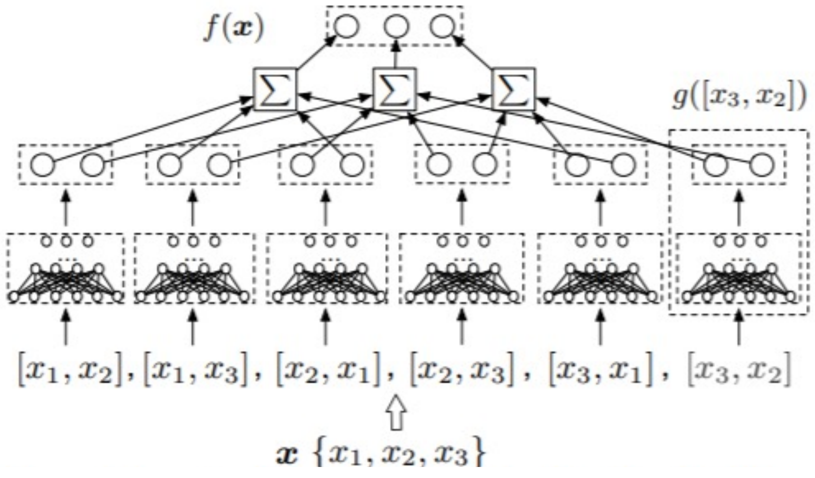
Xi, Yunjia, Weiwen Liu, Xinyi Dai, Ruiming Tang, Weinan Zhang, Qing Liu, Xiuqiang He, and Yong Yu. "Utility-oriented re-ranking with counterfactual context." *arXiv preprint arXiv:2110.09059* (2021).

Beyond Evaluator-Generator Paradigm

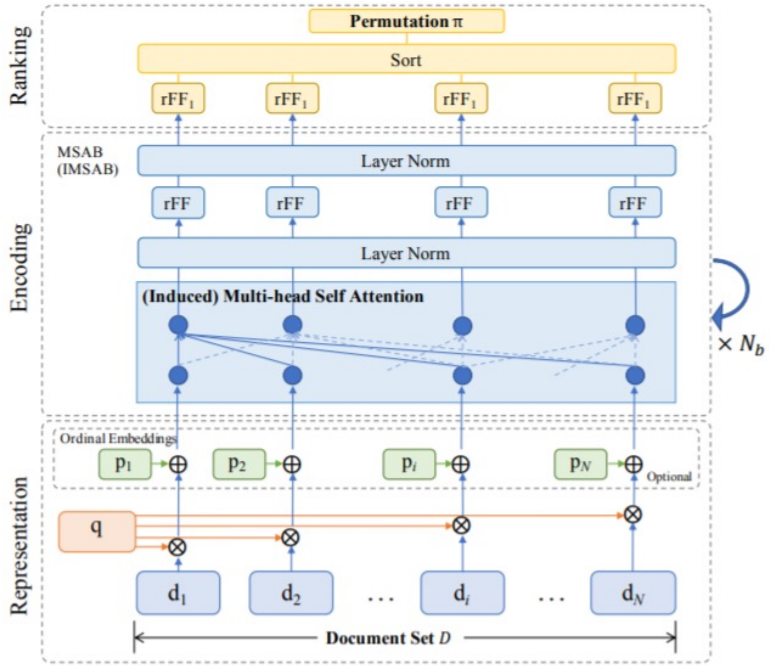
- Design a permutation-invariant model
  - Any permutation of the inputs would not change the output ranking

$$f(\{x_1, \ldots, x_M\}) = f(\{x_{\pi(1)}, \ldots, x_{\pi(M)}\})$$

- GSF
  - DNN on all feasible permutations
- SetRank
  - Multi-head self-attention



GSF

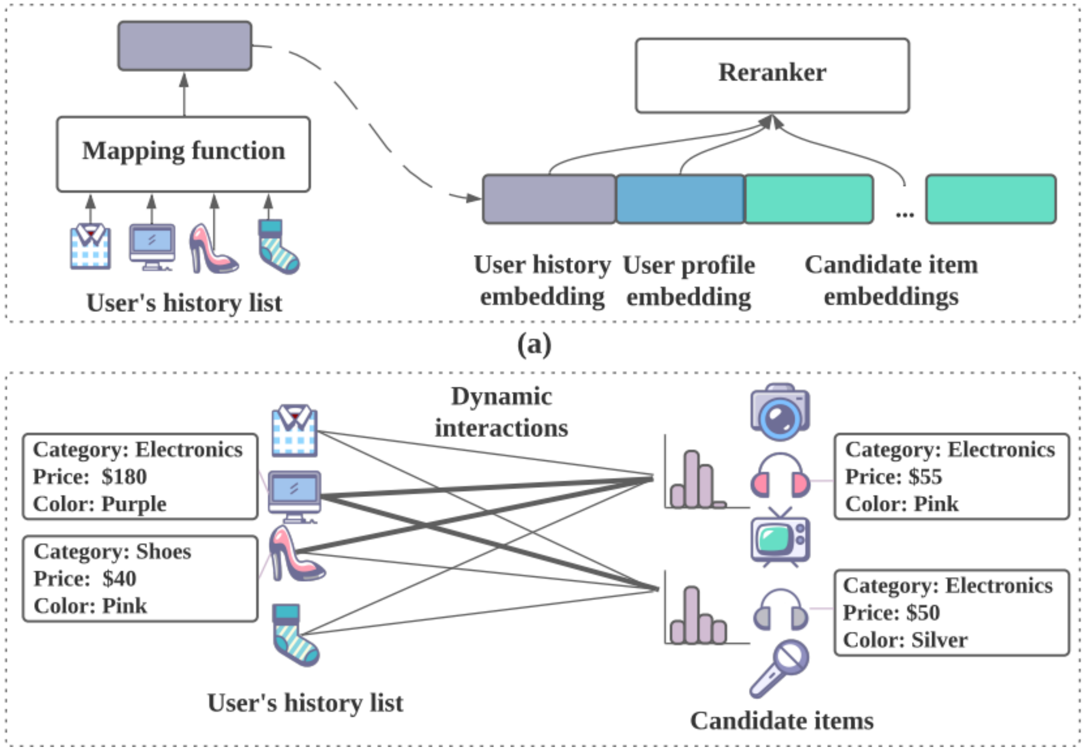


SetRank

Qingyao Ai, Xuanhui Wang, et al. Learning groupwise multivariate scoring functions using deep neural networks. In ICTIR, 2019.
Liang Pang, Jun Xu, et al. Setrank: Learning a permutation-invariant ranking model for information retrieval. In SIGIR, 2020.

Multi-Level Interaction Reranking with User Behavior History

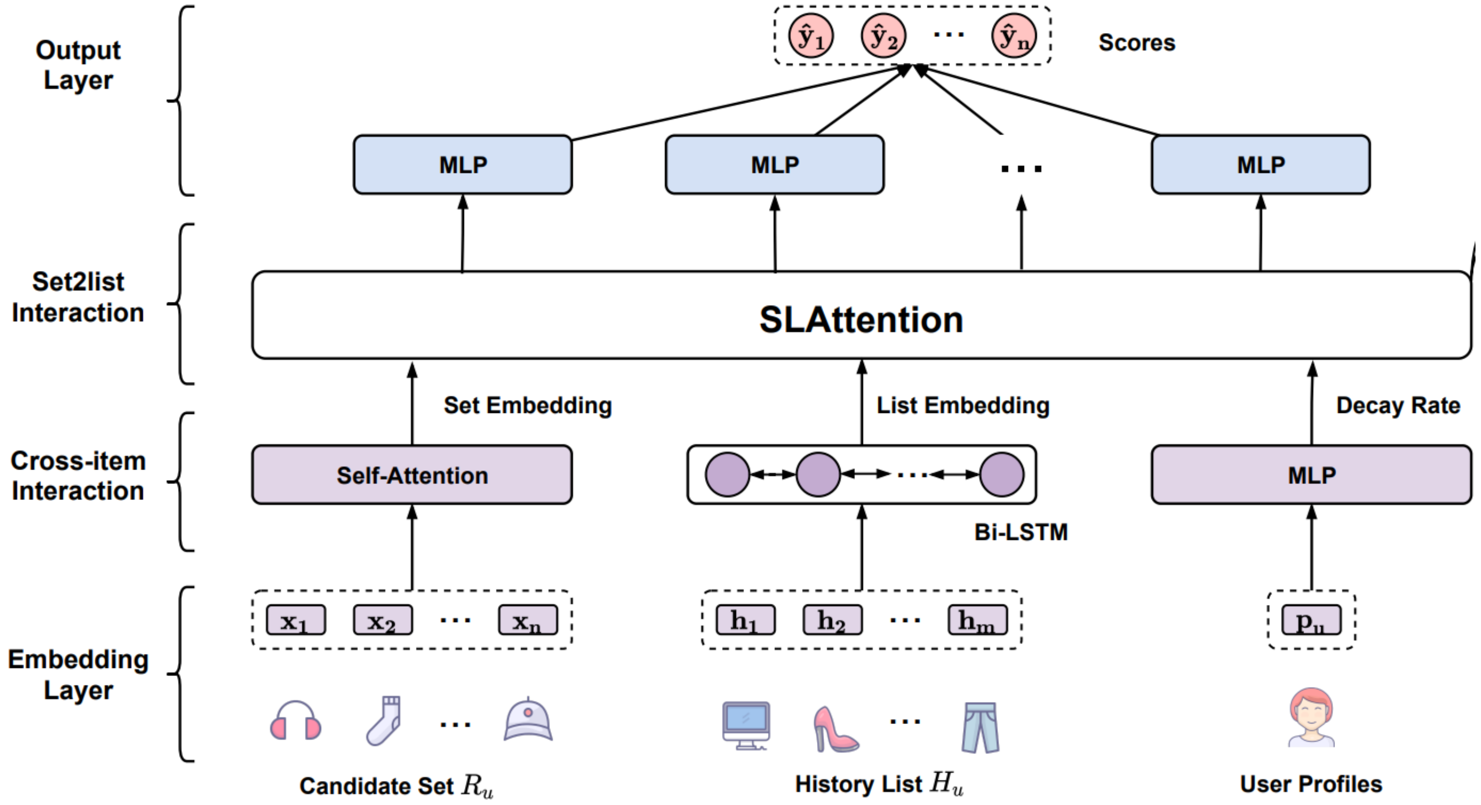
- Exploit personalized preferences from **user behavior history** in a **permutation-invariant** model
  - The items in history contribute **differently** to reranking.
    - Dynamic interaction between user history and candidate items
  - Users' interests are evolving over time.
    - Long-term interest
    - Short-term interest



Yunjia Xi, Weiwen Liu, Jieming Zhu, Xilong Zhao, Xinyi Dai, Ruiming Tang, Weinan Zhang, Rui Zhang, and Yong Yu. "Multi-Level Interaction Reranking with User Behavior History." In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 814-824. 2022.

Multi-Level Interaction Reranking with User Behavior History: **Framework**

- Candidates ⇒ set, user behavior history ⇒ list
- **Cross-item interaction**
  - Intra-set：self-attention
  - Intra-list：Bi-LSTM
- **Set2List interaction**
  - SLAttention



Yunjia Xi, Weiwen Liu, Jieming Zhu, Xilong Zhao, Xinyi Dai, Ruiming Tang, Weinan Zhang, Rui Zhang, and Yong Yu. "Multi-Level Interaction Reranking with User Behavior History." In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 814-824. 2022.

Multi-Level Interaction Reranking with User Behavior History: **SLAttention**

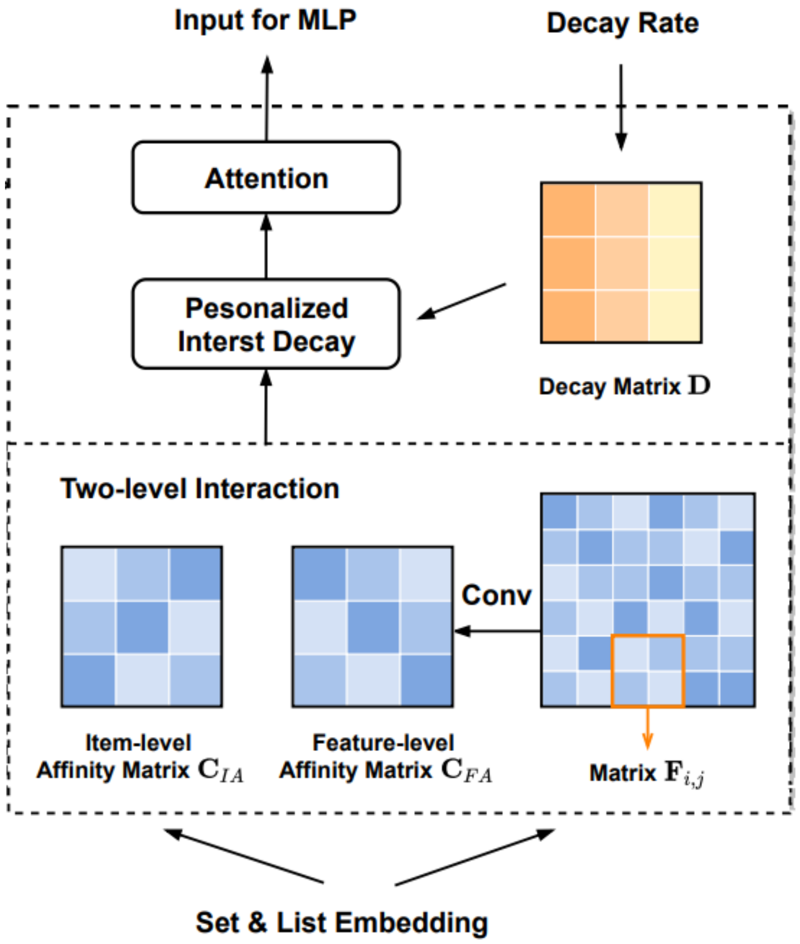
- Dynamic interaction between set and list
- Set embedding S, list embedding L
- Item & feature-level affinity matrix

$$C_{IA} = \tanh(SW_{IA}L^T)$$

$$F_{i,j} = \tanh\left(E_S^i W_{FA} (E_L^j)^T\right)$$

$$C_{FA}(i,j) = \sum_{s=1}^{k}\sum_{t=1}^{k} F_{i,j}(s,t) W_c(s,t)$$

$$C_A = C_{IA} + C_{FA}$$



Yunjia Xi, Weiwen Liu, Jieming Zhu, Xilong Zhao, Xinyi Dai, Ruiming Tang, Weinan Zhang, Rui Zhang, and Yong Yu. "Multi-Level Interaction Reranking with User Behavior History." In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 814-824. 2022.

Multi-Level Interaction Reranking with User Behavior History: **SLAttention**

- Personalized interest decay

$$\theta_u = g(p_u), \qquad d = e^{-\theta_u t_u}$$

$$C = C_A + C_A \odot D$$

- Attention

$$Q_S = \tanh\left(SW_s + C(LW_l)\right)$$

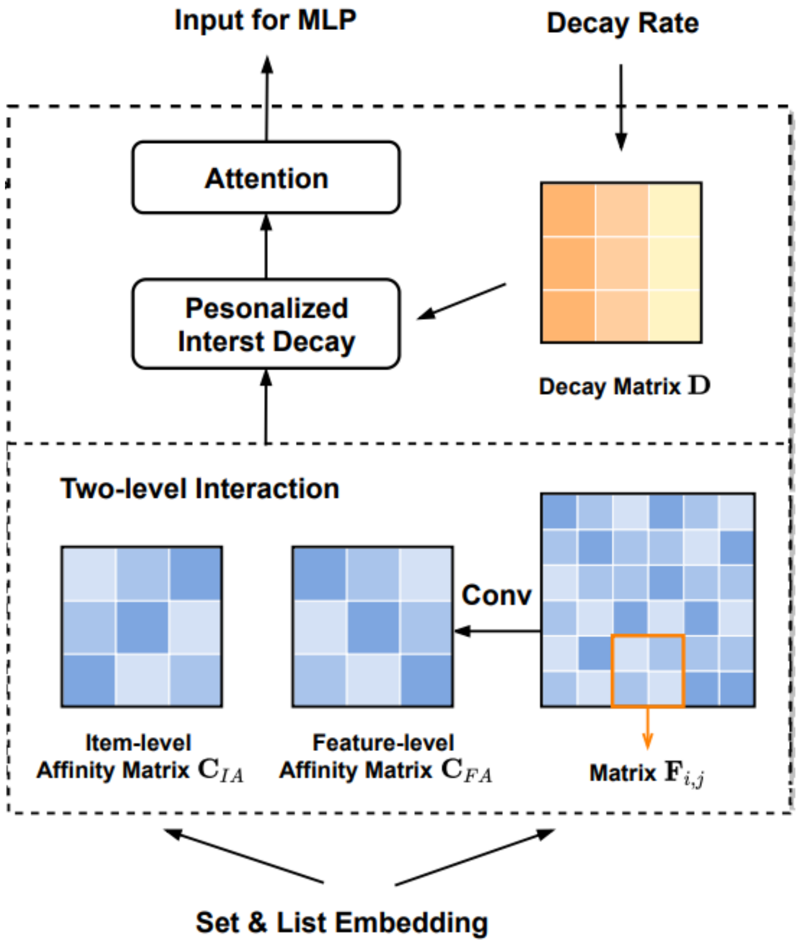
$$Q_L = \tanh(SW_s C)$$

$$A_S = softmax(Q_S)$$

$$A_L = softmax(Q_L)$$

$$\hat{S} = A_S S, \qquad \hat{L} = A_L L$$

- The model is proved to be permutation-invariant



Yunjia Xi, Weiwen Liu, Jieming Zhu, Xilong Zhao, Xinyi Dai, Ruiming Tang, Weinan Zhang, Rui Zhang, and Yong Yu. "Multi-Level Interaction Reranking with User Behavior History." In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 814-824. 2022.

Multi-Level Interaction Reranking with User Behavior History: **Experiment**

- Performance on public two datasets: PRM public and Ad.
  - Ranking metrics: MAP, NDCG, deNDCG
  - Utility metric: Utility

Ranker	Reranker	PRM Public								Ad							
		@10				@20				@5				@10			
		MAP	NDCG	deNDCG	Utility	MAP	NDCG	deNDCG	Utility	MAP	NDCG	deNDCG	Utility	MAP	NDCG	deNDCG	Utility
DIN	initial	0.1929	0.2290	0.2298	1.2289	0.1976	0.3318	0.3324	1.7986	0.5930	0.6660	0.6654	2.2416	0.6028	0.6941	0.6940	2.3574
	MIDNN	0.2986	0.3399	0.3124	1.3267	0.2907	0.4200	0.3965	1.8418	0.5991	0.6705	0.6710	2.2174	0.6093	0.6990	0.6994	2.3345
	DLCM	0.3002	0.3422	0.3146	1.3431	0.2919	0.4227	0.3991	1.8426	0.5998	0.6715	0.6715	2.3126	0.6094	0.6992	0.6995	2.4257
	GSF	0.2989	0.3402	0.3127	1.3283	0.2909	0.4199	0.3964	1.8418	0.5995	0.6710	0.6713	2.2341	0.6097	0.6993	0.6996	2.3516
	PRM	0.3026	0.3446	0.3161	1.3423	0.2940	0.4252	0.4011	1.8653	0.6014	0.6722	0.6725	2.2350	0.6117	0.7006	0.7011	2.3493
	SetRank	0.3003	0.3413	0.3118	1.3192	0.2919	0.4207	0.3951	1.8320	0.6007	0.6718	0.6719	2.2457	0.6101	0.6995	0.6997	2.3624
	MIR	**0.3087***	**0.3511***	**0.3239***	**1.3906***	**0.2989***	**0.4310***	**0.4078***	**1.9064***	**0.6068***	**0.6768***	**0.6771***	**2.3807***	**0.6164***	**0.7044**	**0.7048***	**2.4918***
SVMRank	initial	0.1746	0.2057	0.2093	1.1572	0.1815	0.3079	0.3110	1.7176	0.5864	0.6607	0.6603	2.1978	0.5964	0.6889	0.6888	2.3142
	MIDNN	0.2982	0.3394	0.3113	1.3276	0.2905	0.4193	0.3948	1.8409	0.5975	0.6694	0.6697	2.2192	0.6074	0.6972	0.6975	2.3353
	DLCM	0.2975	0.3383	0.3094	1.3120	0.2896	0.4185	0.3933	1.8293	0.5991	0.6708	0.6712	2.3236	0.6090	0.6983	0.6987	2.4157
	GSF	0.2990	0.3404	0.3120	1.3287	0.2910	0.4200	0.3952	1.8417	0.5987	0.6702	0.6704	2.2354	0.6085	0.6980	0.6983	2.3486
	PRM	0.3005	0.3414	0.3116	1.3175	0.2919	0.4210	0.3951	1.8328	0.5997	0.6705	0.6705	2.1679	0.6098	0.6988	0.6990	2.2842
	SetRank	0.3002	0.3418	0.3120	1.3211	0.2920	0.4209	0.3949	1.8320	0.5980	0.6698	0.6701	2.3118	0.6079	0.6975	0.6979	2.4237
	MIR	**0.3084***	**0.3514***	**0.3230***	**1.3866***	**0.2993***	**0.4308***	**0.4059***	**1.8989***	**0.6056***	**0.6760***	**0.6765***	**2.3683***	**0.6151***	**0.7029**	**0.7033***	**2.4776***
LambdaMART	initial	0.1820	0.2139	0.2158	1.1569	0.1879	0.3155	0.3174	1.7188	0.5897	0.6633	0.6629	2.1783	0.5997	0.6915	0.6915	2.2948
	MIDNN	0.2984	0.3396	0.3127	1.3269	0.2906	0.4196	0.3967	1.8427	0.5979	0.6697	0.6704	2.2329	0.6077	0.6975	0.6980	2.3481
	DLCM	0.2984	0.3394	0.3118	1.3149	0.2906	0.4190	0.3954	1.8295	0.5995	0.6710	0.6712	2.2801	0.6093	0.6988	0.6990	2.3739
	GSF	0.2988	0.3400	0.3130	1.3293	0.2909	0.4200	0.3969	1.8441	0.5991	0.6706	0.6710	2.2735	0.6092	0.6986	0.6990	2.3873
	PRM	0.3002	0.3415	0.3129	1.3156	0.2919	0.4210	0.3966	1.8299	0.6004	0.6714	0.6712	2.2171	0.6107	0.6996	0.6997	2.3327
	SetRank	0.2999	0.3413	0.3132	1.3210	0.2917	0.4206	0.3966	1.8333	0.6001	0.6716	0.6715	2.2789	0.6098	0.6991	0.6993	2.3923
	MIR	**0.3083***	**0.3511***	**0.3247***	**1.3907***	**0.2991***	**0.4301***	**0.4073***	**1.8998***	**0.6060***	**0.6762***	**0.6765***	**2.3685***	**0.6157***	**0.7037**	**0.7042***	**2.4799***

$*$ denotes statistically significant improvement (measured by t-test with p-value < 0.05) over the best baseline.

Yunjia Xi, Weiwen Liu, Jieming Zhu, Xilong Zhao, Xinyi Dai, Ruiming Tang, Weinan Zhang, Rui Zhang, and Yong Yu. "Multi-Level Interaction Reranking with User Behavior History." In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 814-824. 2022.

Multi-Level Interaction Reranking with User Behavior History: **Experiment**

- Performance on a real-world dataset: App Store

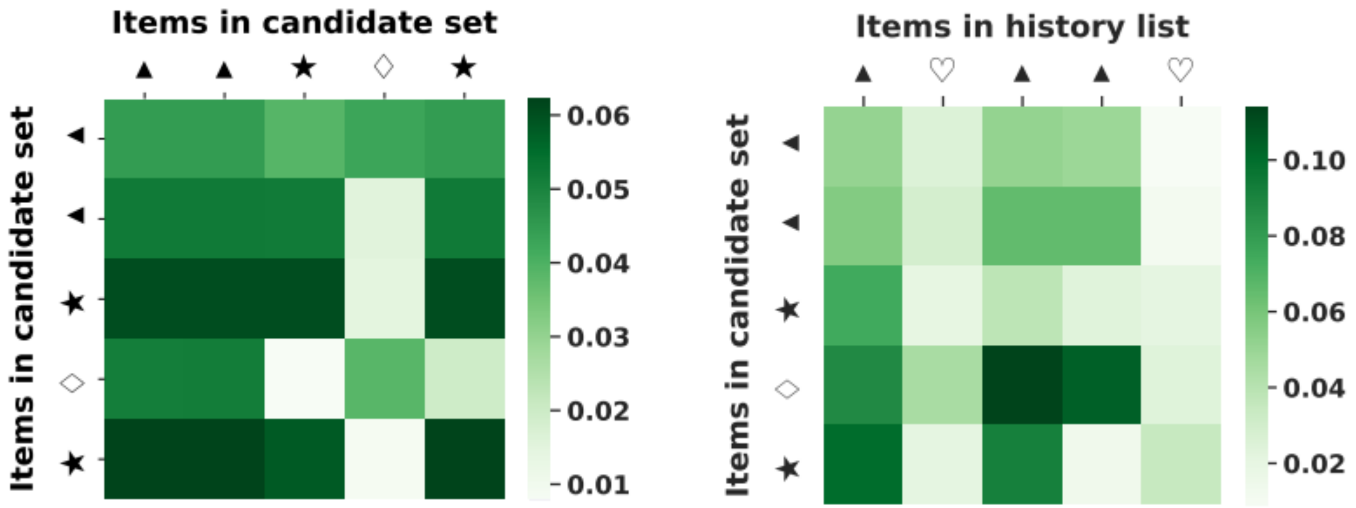
Model	@5				@10			
	MAP	NDCG	deNDCG	Utility	MAP	NDCG	deNDCG	Utility
init	0.1855	0.3549	0.3474	2.4671	0.1809	0.4260	0.4200	3.3646
MIDNN	0.2352	0.4349	0.4340	3.5379	0.2293	0.5014	0.5008	4.6408
DLCM	0.3205	0.5074	0.5112	3.9615	0.3145	0.5588	0.5623	4.8690
GSF	0.2271	0.4253	0.4249	3.4690	0.2213	0.4941	0.4942	4.6046
PRM	0.3281	0.5132	0.5166	3.9945	0.3222	0.5662	0.5685	4.9185
SetRank	0.2591	0.4537	0.4569	3.6234	0.2533	0.5168	0.5194	4.6845
MIR	**0.3449***	**0.5301***	**0.5337***	**4.0964***	**0.3396***	**0.5815***	**0.5838***	**5.0014***

$*$ denotes statistically significant improvement (measured by t-test with p-value $<$ 0.05) over the best baseline.

Yunjia Xi, Weiwen Liu, Jieming Zhu, Xilong Zhao, Xinyi Dai, Ruiming Tang, Weinan Zhang, Rui Zhang, and Yong Yu. "Multi-Level Interaction Reranking with User Behavior History." In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 814-824. 2022.

Multi-Level Interaction Reranking with User Behavior History: **Experiment**

- Visualization of the attention coefficient obtained in SLAttention
  - The weights of the same type of items are similar in most cases, such as triangular items.



(a) A_S (b) A_L

Yunjia Xi, Weiwen Liu, Jieming Zhu, Xilong Zhao, Xinyi Dai, Ruiming Tang, Weinan Zhang, Rui Zhang, and Yong Yu. "Multi-Level Interaction Reranking with User Behavior History." In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 814-824. 2022.

Qualitative Model Comparison

	Listwise context modeling	Optimization	Personalization	Complexity
DLCM [2018]	GRU	AttRank	NP	$\mathcal{O}(n)$
MiDNN[2018]	LSTM	CE	D	$\mathcal{O}(n)$
ListCVAE [2018]	CVAE	KL	NP	$\mathcal{O}(n)$
Seq2Slate [2018]	PointerNet	CE	NP	$\mathcal{O}(n^2)$
HRM [2019]	Similarity	Hinge	D	$\mathcal{O}(hn + h^2)$
PRM [2019]	Self-attention	CE	D	$\mathcal{O}(n^2)$
IRGPR [2020b]	GNN	BPR	M	$\mathcal{O}(n)$
PFRN [2020]	Self-attention	CE	D	$\mathcal{O}(n^2 + h^2)$
PivotCVAE [2021]	CVAE	KL	NP	$\mathcal{O}(n)$
Raise [2022]	Self-attention	CE	M	$\mathcal{O}(n^2)$
PEAR [2022]	Self-/cross-attention	CE	D	$\mathcal{O}((n + h)^2)$
GSF [2019]	DNN	CE	NP	$O(\frac{mn!}{(n-m)!})$
SEG [2019]	E: BiGRU G: GRU	MSE/Q-learning	D	$\mathcal{O}(n^2)$
SetRank [2020]	Self-attention	AttRank	NP	$\mathcal{O}(n^2)$
CDIA [2020]	E: LSTM G: LSTM	Policy gradient	D	$\mathcal{O}(n^2)$
GCR [2020]	E: BiGRU+attention G: GRU	PPO-exploration	D	$\mathcal{O}(n^2)$
PRS [2021a]]	E: BiLSTM G: Beam search	—	D	$\mathcal{O}(n^2)$
GRN [2021b]	E: BiLSTM+attention G: GRU+attention+ PointerNet	Policy gradient	D	$\mathcal{O}(n^2)$
CRUM [2021]	E: BiLSTM+GNN G: MLP	LambdaLoss	D	$\mathcal{O}(n)$
EGRerank [2021]	E: LSTM G: LSTM	PPO	D	$\mathcal{O}(n^2)$
MIR [2022]	BiLSTM+attention	CE	D	$\mathcal{O}((n + h)^2)$

Quantitative Model Comparison

- LibRerank: a neural re-ranking library to automates the re-ranking experimentation

	Ad				PRM Public			
	MAP@5	NDCG@5	MAP@10	NDCG@10	MAP@10	NDCG@10	MAP@20	NDCG@20
Init [2010]	0.6037	0.6840	0.6075	0.6990	0.1842	0.2178	0.1901	0.3202
MiDNN [2018]	0.6080	0.6876	0.6117	0.7021	0.3069	0.3482	0.2977	0.4265
GSF [2019]	0.6090	0.6883	0.6126	0.7028	0.3060	0.3459	0.2968	0.4241
EGRerank [2021]	0.6092	0.6890	0.6126	0.7029	0.3075	0.3502	0.2985	0.4286
DLCM [2018]	0.6126	0.6914	0.6162	0.7055	0.3082	0.3500	0.2991	0.4287
SetRank [2020]	0.6132	0.6917	0.6168	0.7060	0.3094	0.3515	0.3002	0.4297
PRM [2019]	0.6140	0.6923	0.6178	0.7066	0.3096	0.3516	0.3003	0.4301

LibRerank

https://github.com/LibRerank-Community/LibRerank

Outline

- Introduction
  - Multi-stage recommender systems
  - Neural re-ranking
- Single objective: Accuracy oriented
  - Learning by observed signals
  - Learning by counterfactual signals
  - LibRerank library
- Multi-objective
  - Diversity-aware re-ranking
  - Fairness-aware re-ranking
- Emerging applications
- Summary

Diversity-aware Re-ranking

- Diversity
  - Measure the dissimilarity or topic coverage of a re-ranking list
- Accuracy-diversity tradeoff
  - Trade-off parameter
  - Optimize a specific metric that combines accuracy and diversity like α-NDCG
- Implicit approach
  - **Query subtopic is not available at inference**
  - M^2-DIV [Feng et al., 2018], DALETOR [Yan et al., 2021]
- Explicit approach
  - **Query subtopic is assumed to be available at inference**
  - DSSA [Jiang et al., 2017], DVGAN [Liu et al., 2020], DESA [Qin et al., 2020]…

Diversity-aware Re-ranking: DALETOR

- Motivation
  - Directly optimize the diversity-aware metric by a differentiable loss
  - “Score-and-sort” instead of “next document”
- Major solution
  - Approximate the diversity-aware metric in a soft version
  - Propose a listwise scoring function

Diversification-Aware Learning to Rank using Distributed Representation. Yuan et al. WWW 2021.

Diversity-aware Re-ranking: DALETOR

- Differentiable approximate diversity-aware metric as training objective (e.g.,α-NDCG)
  - Original format

$$\alpha\text{-DCG} = \sum_{i=1}^{n} \sum_{l=1}^{m} \frac{y_{il}(1-\alpha)^{c_{li}}}{\log_2(1+r_i)}, \qquad c_{li} = \sum_{j:r_j \le r_i} y_{jl}\,. \qquad \alpha\text{-NDCG} = \frac{\alpha\text{-DCG}}{\alpha\text{-DCG}_{\text{opt}}}.$$

  - Transformed format

$$r_i = 1 + \sum_j \mathbb{I}_{s_j > s_i}$$

$$c_{li} = \sum_j y_{jl} \mathbb{I}_{s_j > s_i},$$

Indicator function approximation

$$\text{sigmoid}(x) = \frac{1}{1 + \exp(-x/T)}$$

$$R_i = 1 + \sum_{j \ne i} \text{sigmoid}\left(\frac{s_j - s_i}{T}\right) = \frac{1}{2} + \sum_j \frac{1}{1 + \exp\left(\frac{s_i - s_j}{T}\right)},$$

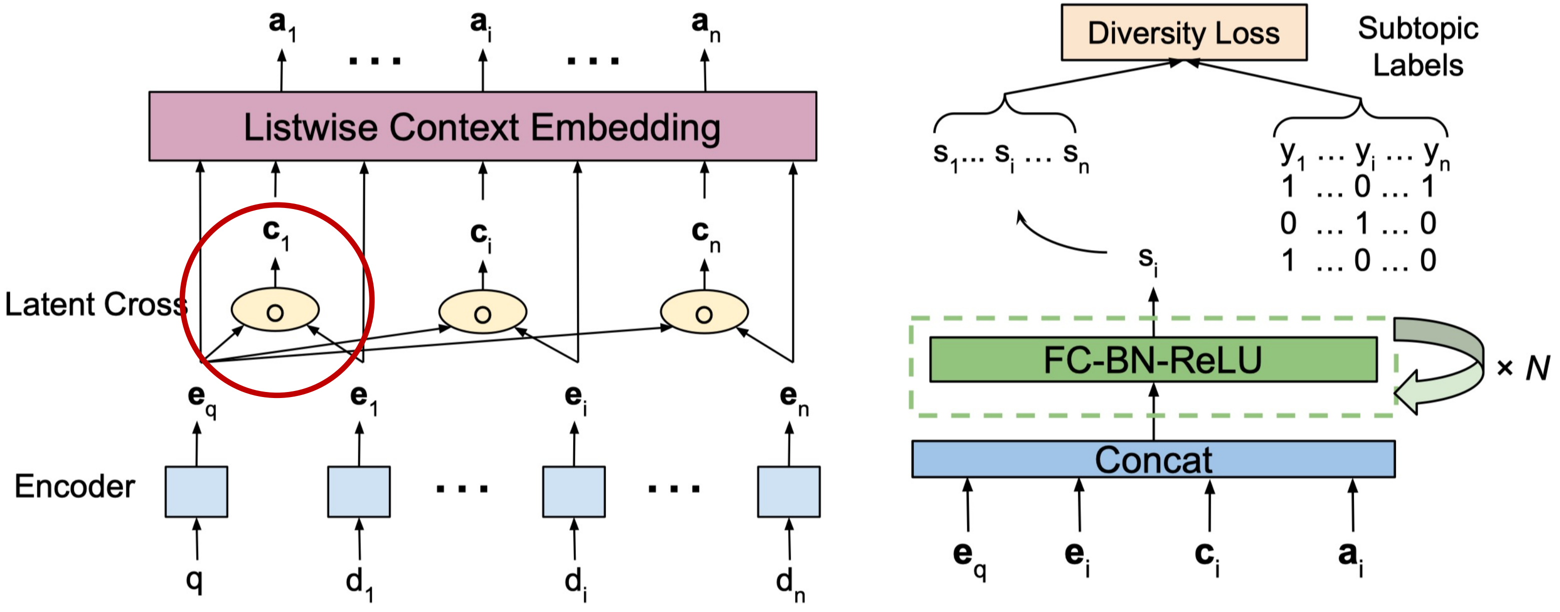
$$C_{li} = \sum_{j \ne i} y_{jl} \cdot \text{sigmoid}\left(\frac{s_j - s_i}{T}\right) = \sum_j \frac{y_{jl}}{1 + \exp\left(\frac{s_i - s_j}{T}\right)} - \frac{y_{il}}{2}$$

$$\mathcal{L}_{\alpha\text{-DCG}}(\{s_i^q\}) = -\frac{1}{|Q|} \sum_{q \in Q} \sum_{i=1}^{n} \sum_{l=1}^{m} \frac{y_{il}^q (1-\alpha)^{C_{li}^q}}{\log_2(1 + R_i^q)},$$

Diversification-Aware Learning to Rank using Distributed Representation. Yuan et al. WWW 2021.

Diversity-aware Re-ranking: DALETOR

- Neural Architecture
  - Distributed representation
    - Query-doc cross feature (latent cros $\mathbf{c}_i = \mathbf{e}_i \circ \mathbf{e}_q.$
  - Listwise context embedding
    - Document interaction network (DIN)
  - Output layer



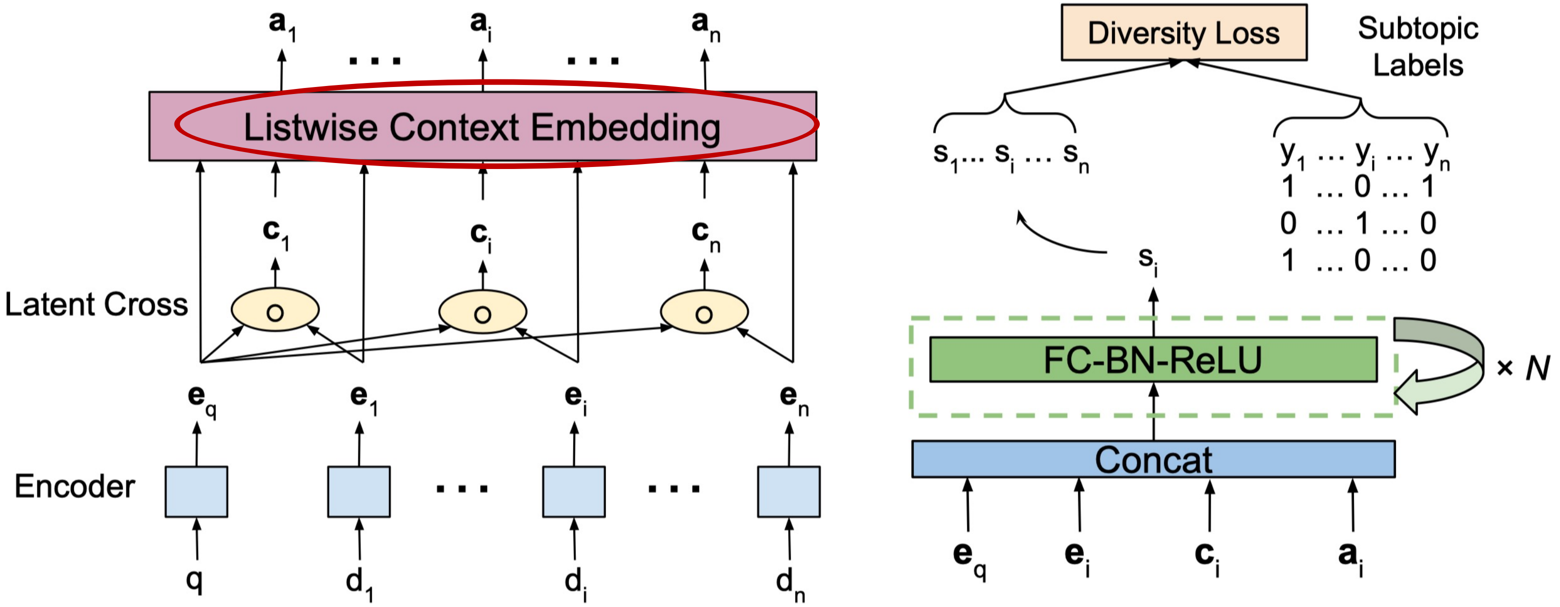
Diversification-Aware Learning to Rank using Distributed Representation. Yuan et al. WWW 2021.

Diversity-aware Re-ranking: DALETOR

- Neural Architecture
  - Distributed representation
    - Query-doc cross feature (latent cross)
  - Listwise context embedding
    - Document interaction network (DIN)
  - Output layer

$$\mathrm{SA}(D) = \mathrm{Softmax}(S(D))V,$$

$$\mathrm{MHSA}(D) = \mathrm{concat}_{h \in [H]} [\mathrm{SA}_h(D)] W_{\mathrm{out}} + b_{\mathrm{out}},$$



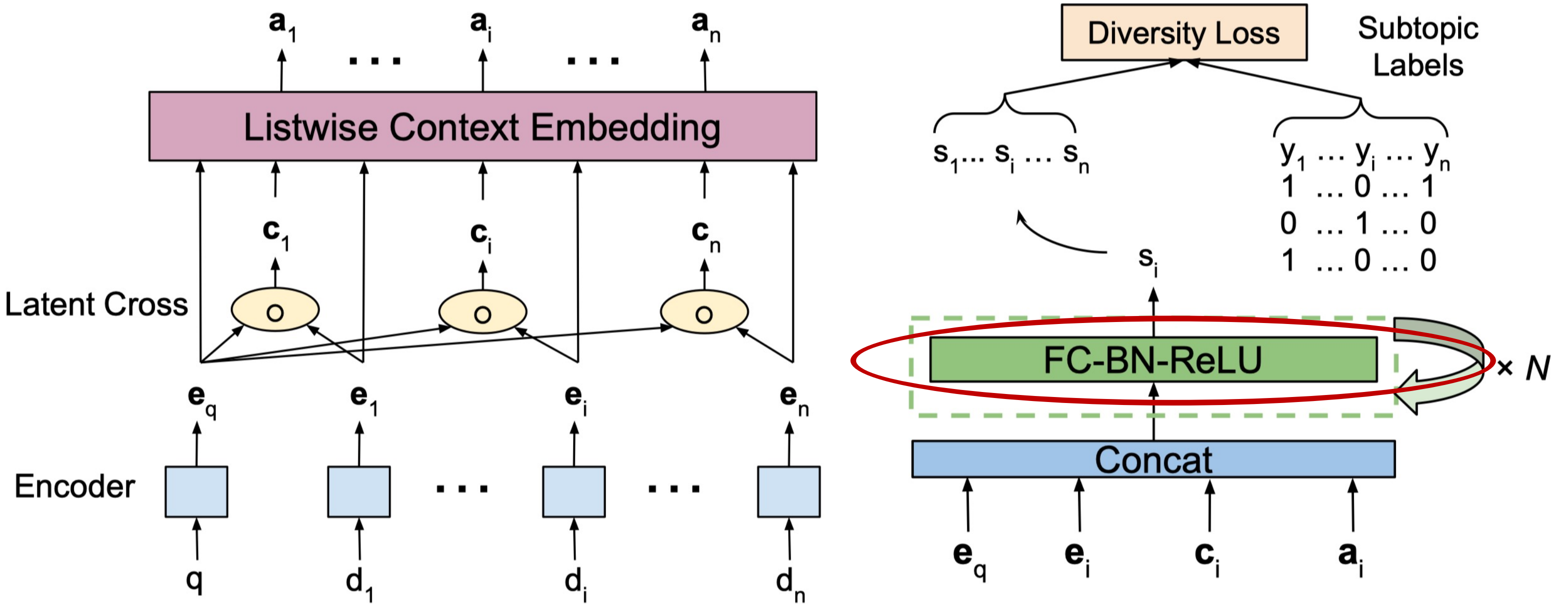
Diversification-Aware Learning to Rank using Distributed Representation. Yuan et al. WWW 2021.

Diversity-aware Re-ranking: DALETOR

- Neural Architecture
  - Distributed representation
    - Query-doc cross feature (latent cross)
  - Listwise context embedding
    - Document interaction network (DIN)
  - Output layer

$$\mathbf{a}_i = \mathrm{DIN}_i(\{\mathrm{concat}(\mathbf{e}_q, \mathbf{e}_j, \mathbf{c}_j)\})$$

$$s_{\mathrm{DALETOR}}(q, \{d_i\}) = \{s(\mathrm{concat}(\mathbf{e}_q, \mathbf{e}_i, \mathbf{c}_i, \mathbf{a}_i))\}.$$



Diversification-Aware Learning to Rank using Distributed Representation. Yuan et al. WWW 2021.

Diversity-aware Re-ranking: DALETOR

- Overall Performance

Method	α-NDCG@5	α-NDCG@10	ERR-IA@5	ERR-IA@10
MMR	0.2753	0.2979	0.2005	0.2309
xQuAD	0.3165	0.3941	0.2314	0.2890
PM-2	0.3047	0.3730	0.2298	0.2814
SVM-DIV	0.3030	0.3699	0.2268	0.2726
R-LTR	0.3498	0.4132	0.2521	0.3011
PAMM	0.3712	0.4327	0.2619	0.3029
NTN-DIV	0.3962	0.4577	0.2773	0.3285
MDP-DIV	0.4189	0.4762	0.2988	0.3494
M^2DIV	0.4429	0.4839	0.3445	0.3658
DNN(softmax)	0.4280	0.4676	0.3293	0.3496
DNN(R-LTR)	0.4149	0.4517	0.3265	0.3454
DNN-LC(α-DCG)	0.4968^*	$\mathbf{0.5322}^*$	0.3868^*	0.4068^*
DIN-LC(α-DCG)	$\mathbf{0.5009}^*$	0.5294^*	$\mathbf{0.3942}^*$	$\mathbf{0.4119}^*$

- Benefits of the α-DCG loss

Method	α-NDCG@5	α-NDCG@10	ERR-IA@5	ERR-IA@10
DNN(R-LTR)	0.4149	0.4517	0.3265	0.3454
DNN(α-DCG)	0.4614^*	0.5005^*	0.3633	0.3838^*
DNN-LC(R-LTR)	0.4451	0.4842	0.3483	0.3690
DNN-LC(α-DCG)	$\mathbf{0.4968}^\dagger$	$\mathbf{0.5322}^\dagger$	$\mathbf{0.3868}^\dagger$	$\mathbf{0.4068}^\dagger$

Diversification-Aware Learning to Rank using Distributed Representation. Yuan et al. WWW 2021.

Diversity-aware Re-ranking: DALETOR

- Latent cross features & listwise scoring

Method	α-NDCG@5	α-NDCG@10	ERR-IA@5	ERR-IA@10
M^2DIV	0.4429	0.4839	0.3445	0.3658
M^2DIV-LC	0.4551	0.4971	0.3509	0.3735
DNN(α-DCG)	0.4614	0.5005	0.3633	0.3838
DNN-LC(α-DCG)	0.4968*	**0.5322***	0.3868	0.4068
DIN(α-DCG)	0.4615	0.5041	0.3582	0.3808
DIN-LC(α-DCG)	**0.5009†**	0.5294	**0.3942†**	**0.4119**
GSF(α-DCG)	0.4568	0.5023	0.3569	0.3802
GSF-LC(α-DCG)	0.4865	0.5219	0.3786	0.4003

- Different approximation of α-DCG

Method	α-NDCG@5	α-NDCG@10	ERR-IA@5	ERR-IA@10
α-DCG (T=0.1)	0.4968	0.5322	**0.3868**	**0.4068**
α-DCG (T=1.0)	0.4811	0.5184	0.3703	0.3912
α-DCG (T=0.01)	0.4715	0.4978	0.3633	0.3799
Gumbel-α-DCG	**0.4970**	**0.5339**	0.3855	0.4066

$$\mathcal{L}_{\text{Gumbel-}\alpha\text{-DCG}}(\{s_i^q\}) = \mathbb{E}_g[\mathcal{L}_{\alpha\text{-DCG}}(\{\beta(s_i^q + g_i)\})],$$

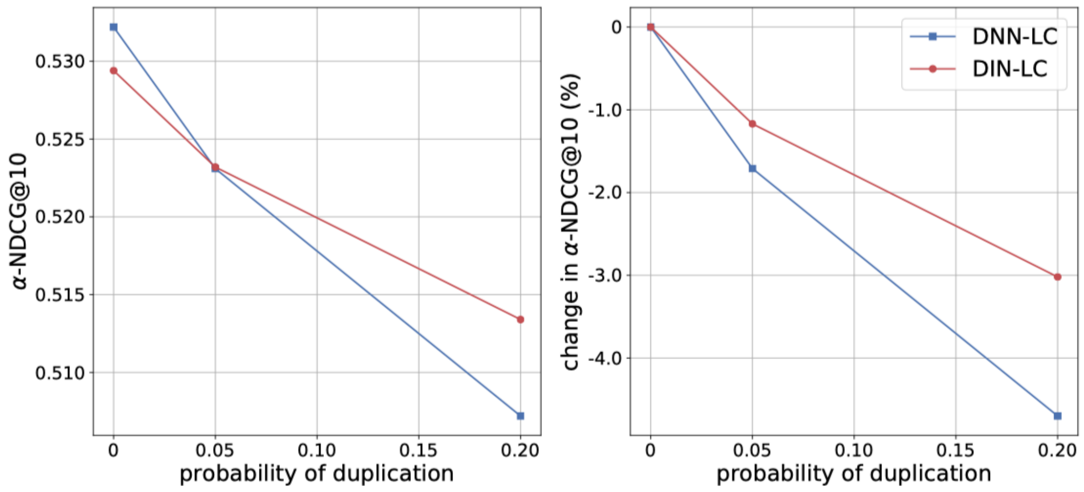
Diversification-Aware Learning to Rank using Distributed Representation. Yuan et al. WWW 2021.

Diversity-aware Re-ranking: DALETOR

- Different SA layers

(L, H, z)	α-NDCG@5	α-NDCG@10	ERR-IA@5	ERR-IA@10
(1, 1, 256)	0.4724	0.5182	0.3679	0.3915
(1, 2, 256)	0.4761	0.5139	0.3706	0.3908
(1, 3, 256)	0.4893	0.5224	0.3801	0.3993
(1, 4, 256)	0.4895	0.5252	0.3810	0.4010
(2, 1, 256)	0.4918	0.5299	0.3842	0.4052
(2, 2, 256)	0.5009	0.5294	0.3942	0.4119
(2, 3, 256)	0.4902	0.5224	0.3800	0.3991
(2, 4, 256)	**0.5066**	**0.5344**	**0.3950**	**0.4122**
(2, 2, 128)	0.4931	0.5295	0.3842	0.4048
(2, 2, 64)	0.4908	0.5245	0.3796	0.3993

- Performance degradation on perturbed dataset



Diversification-Aware Learning to Rank using Distributed Representation. Yuan et al. WWW 2021.

Diversity-aware Re-ranking: DESA

- Motivation
  - Leverage both novelty and relevance of documents at the same time as a whole sequence
  - Model the listwise interactions between the documents
- Major solution
  - Self-attentive network structure

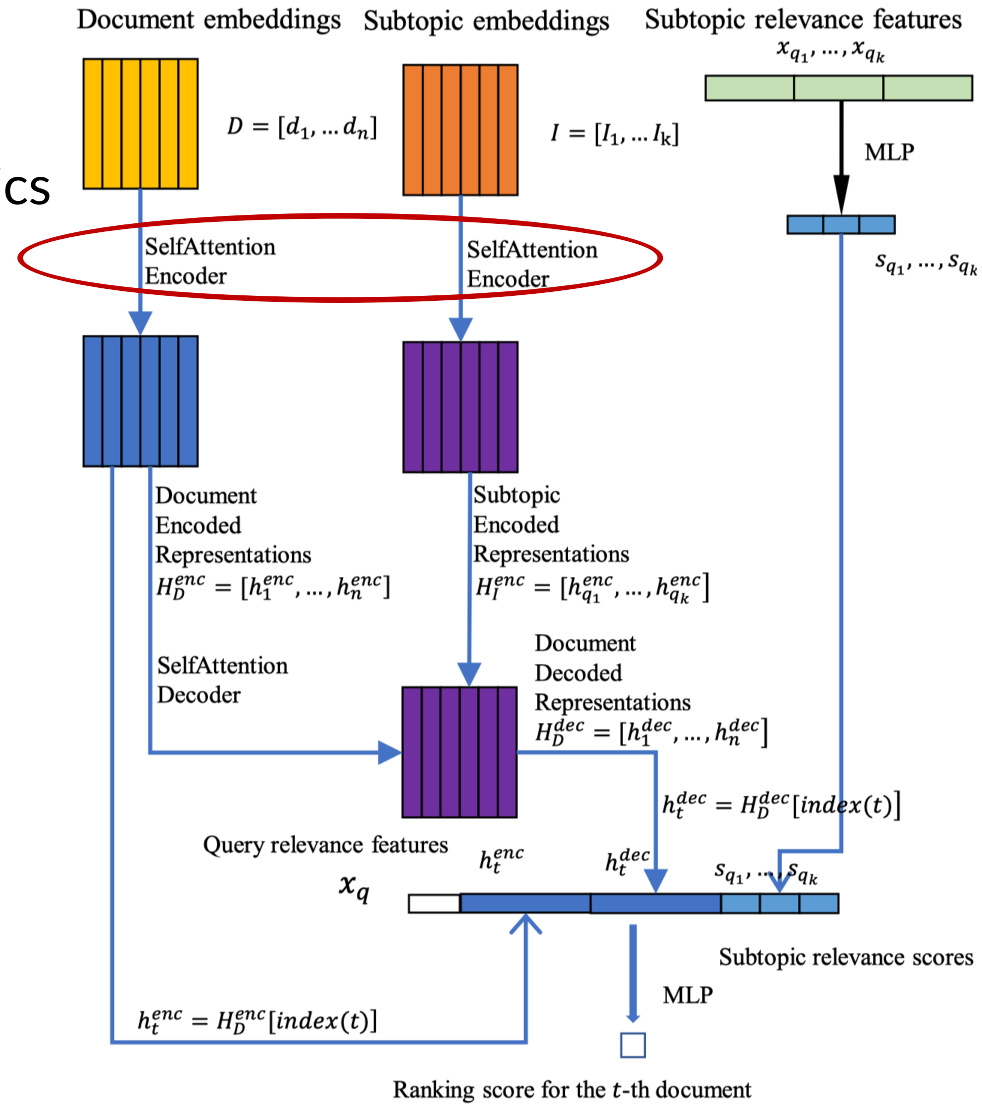
Diversifying Search Results using Self-Attention Network. Qin et al. CIKM 2020.

Diversity-aware Re-ranking: DESA

- Network architecture
  - Document representation (doc2vec)
  - Self-attention encoder (candidates and topics

$$H_D^{\text{enc}} = \text{SelfAttnEnc}(D),$$
$$H_I^{\text{enc}} = \text{SelfAttnEnc}(I),$$

  - Self-attention decoder
  - Subtopic document ranking
  - Final ranking



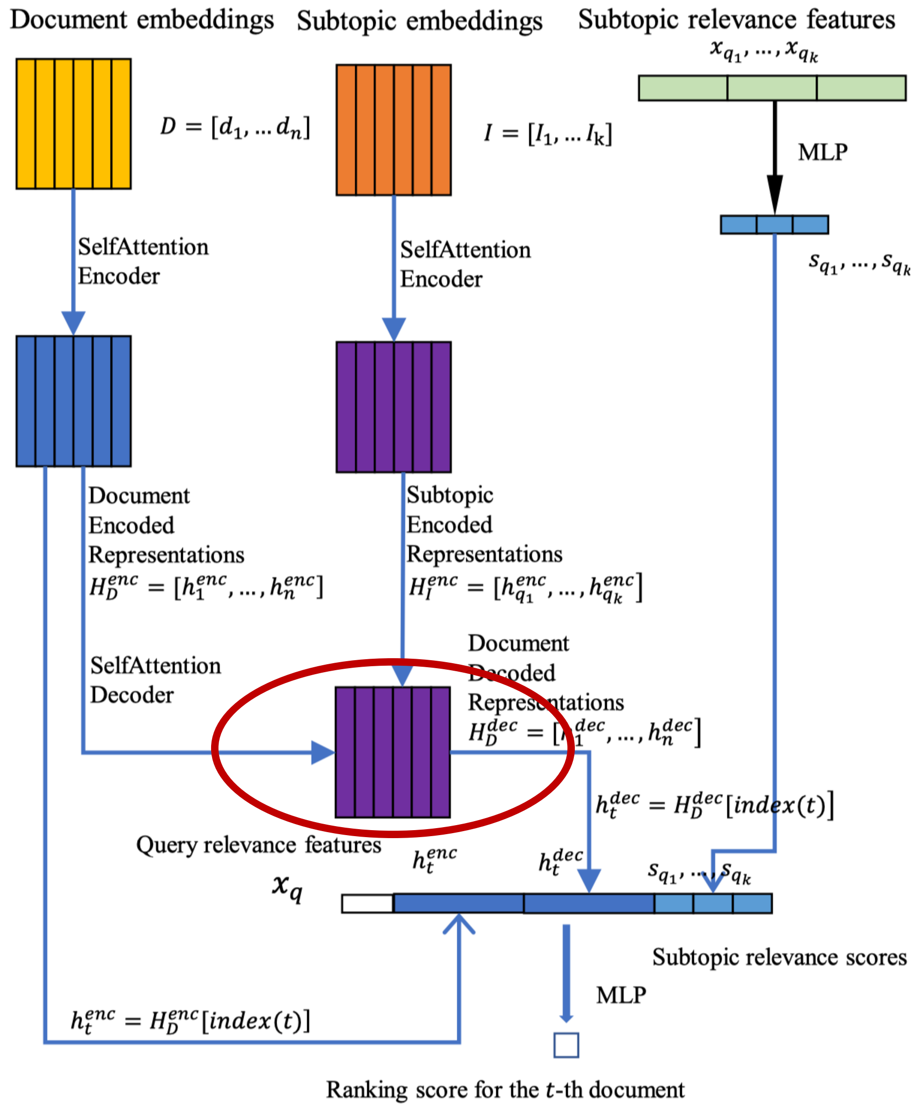
Diversifying Search Results using Self-Attention Network. Qin et al. CIKM 2020.

Diversity-aware Re-ranking: DESA

- Network architecture
  - Document representation (doc2vec)
  - Self-attention encoder (candidates and topics
  - Self-attention decoder

$$H_D^{\mathrm{dec}} = \mathrm{SelfAttnDec}(H_D^{\mathrm{enc}}, H_I^{\mathrm{enc}}),$$
$$h_t^{\mathrm{enc}} = H_D^{\mathrm{enc}}[\mathrm{index(t)}],$$
$$h_t^{\mathrm{dec}} = H_D^{\mathrm{dec}}[\mathrm{index(t)}],$$

  - Subtopic document ranking
  - Final ranking



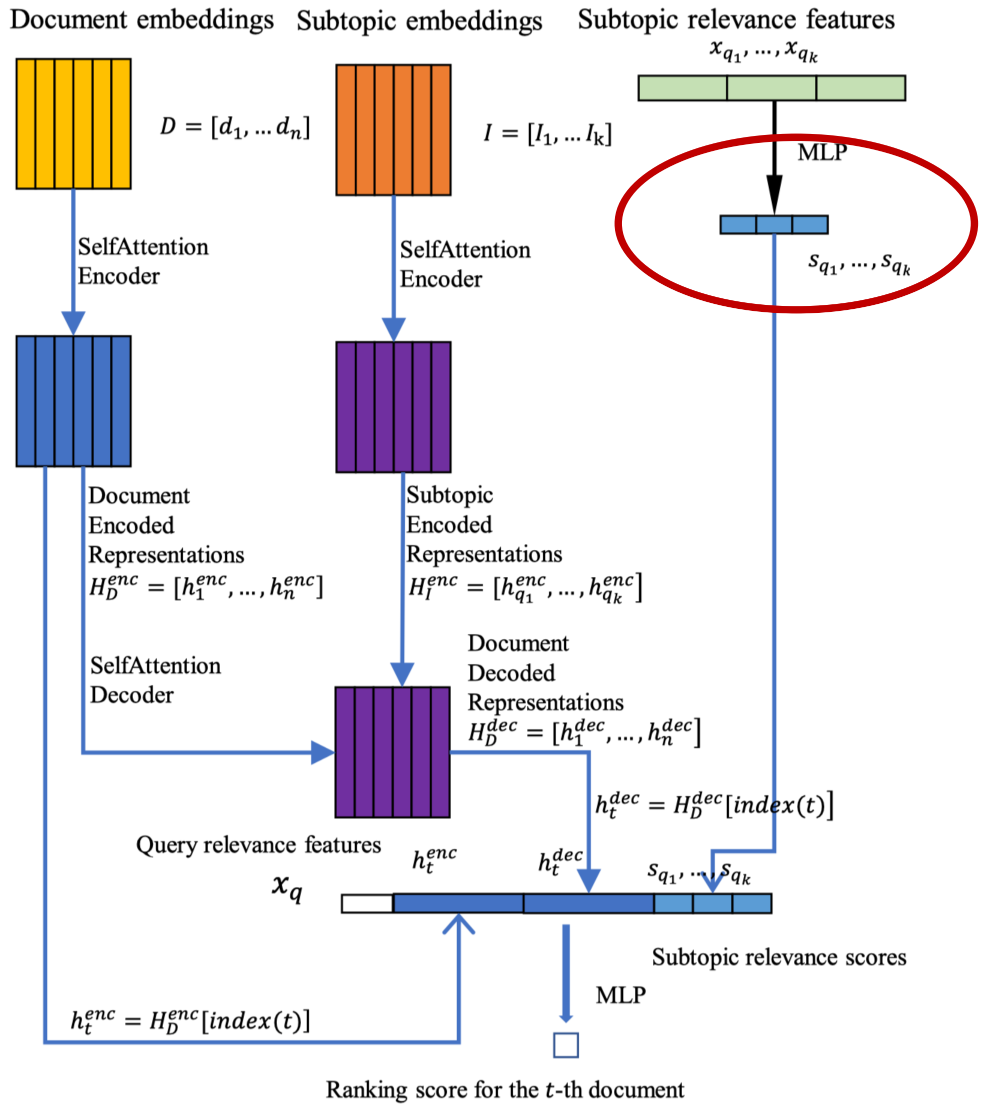
Diversifying Search Results using Self-Attention Network. Qin et al. CIKM 2020.

Diversity-aware Re-ranking: DESA

- Network architecture
  - Document representation (doc2vec)
  - Self-attention encoder (candidates and topics
  - Self-attention decoder
  - Subtopic document ranking

$$s_{q_i} = x_{q_i}^T w_r (i \in [1, k]).$$

  - Final ranking



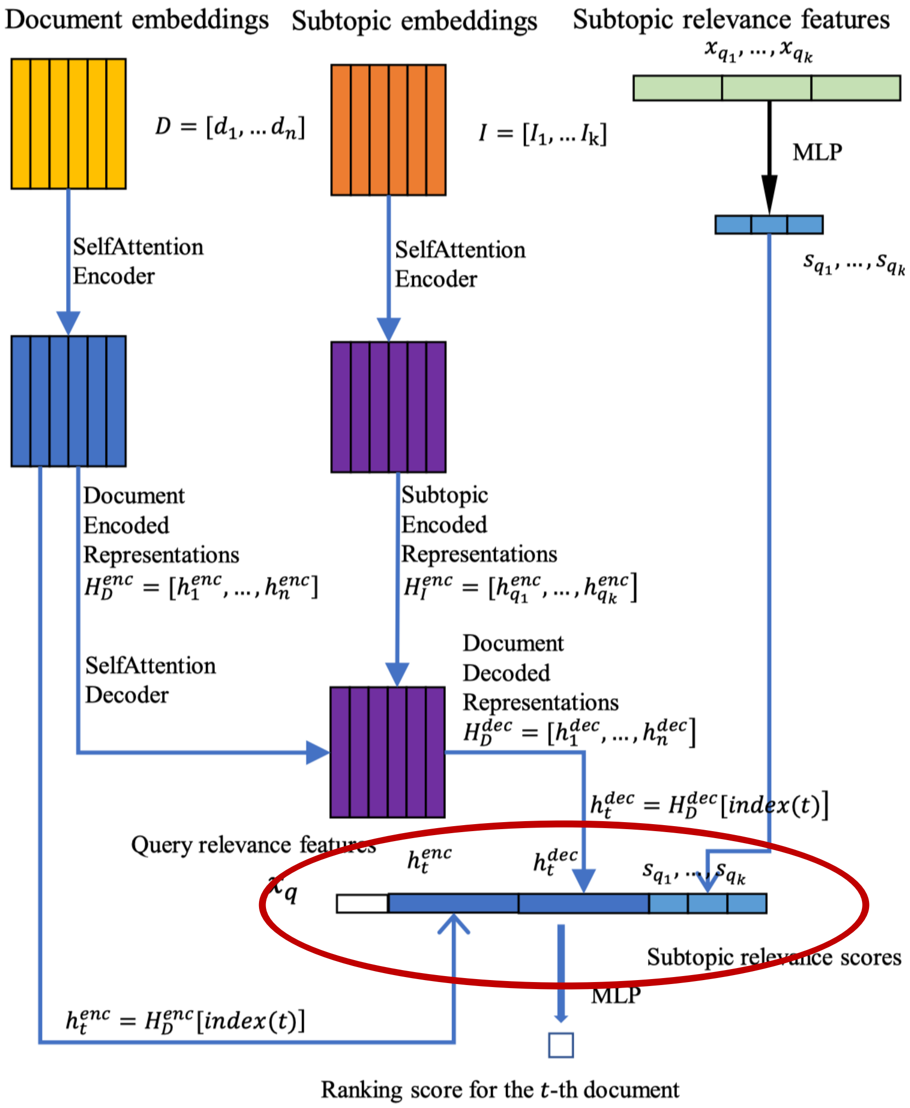
Diversifying Search Results using Self-Attention Network. Qin et al. CIKM 2020.

Diversity-aware Re-ranking: DESA

- Network architecture
  - Document representation (doc2vec)
  - Self-attention encoder (candidates and topics
  - Self-attention decoder
  - Subtopic document ranking
  - Final ranking

$$v_{d_t,q,q_i} = [x_q; h_t^{\text{enc}}; h_t^{\text{dec}}; s_{q_1}, \ldots, s_{q_k}],$$

$$s_t = v_{d_t,q,q_i}^T w_v.$$



Diversifying Search Results using Self-Attention Network. Qin et al. CIKM 2020.

Diversity-aware Re-ranking: DESA

- Training & Optimization
  - Score of ranking list is $s_r = \sum_{i=1}^{|r|} s_i.$
  - List-pairwise sampling

$$Loss = \sum_{q \in Q} \sum_{s \in S_q} |\Delta M| [y_s \log(P(r_1, r_2)) + (1 - y_s) \log(1 - P(r_1, r_2))]$$

  - The context-based pairwise loss function

$$s_{r_1} - s_{r_2} = s_{d_1} - s_{d_2},$$
$$P(r_1, r_2) = P(d_1, d_2).$$

$$Loss = \sum_{q \in Q} \sum_{[C,(d_1,d_2)] \in S_q} |\Delta M| \mathrm{LogLoss}(\mathrm{P}(\mathrm{d}_1, \mathrm{d}_2)).$$

Diversifying Search Results using Self-Attention Network. Qin et al. CIKM 2020.

Diversity-aware Re-ranking: DESA

- Overall performance

Methods	ERR-IA	α-nDCG	NRBP	Pre-IA	S-rec
Lemur	.271	.369	.232	.153	.621
xQuAD	.317	.413	.284	.161	.622
PM2	.306	.411	.267	.169	.643
HxQuAD	.326	.421	.294	.158	.629
HPM2	.317	.420	.279	.172	.645
R-LTR	.303	.403	.267	.164	.631
PAMM	.309	.411	.271	.168	.643
R-LTR-NTN	.312	.415	.272	.166	.644
PAMM-NTN	.311	.417	.272	.170	.648
DSSA (doc2vec)	.350	.452	.318	.184	.645
DESA	**.363**★	**.464**★	**.332**★	.184	**.653**★

- Effects of different settings

Settings	ERR-IA	α-nDCG	NRBP	Pre-IA	S-rec
$L_{enc}=1, L_{dec}=1$	.357	.457	.324	.183	.650
$L_{enc}=2, L_{dec}=1$	**.364**	**.464**	**.332**	**.184**	**.653**
$L_{enc}=3, L_{dec}=1$	.355	.455	.323	.182	.654
$L_{enc}=1, L_{dec}=2$	.361	.462	.329	.182	.658
$L_{enc}=2, L_{dec}=2$	.358	.460	.324	.180	.658
No Subtopics	.344	.445	.311	.177	.648
Relevance Scores	.357	.458	.326	.183	.653
Encoded Subtopics	**.364**	**.464**	**.332**	**.184**	**.653**
Original Subtopics	.349	.453	.313	.180	.655

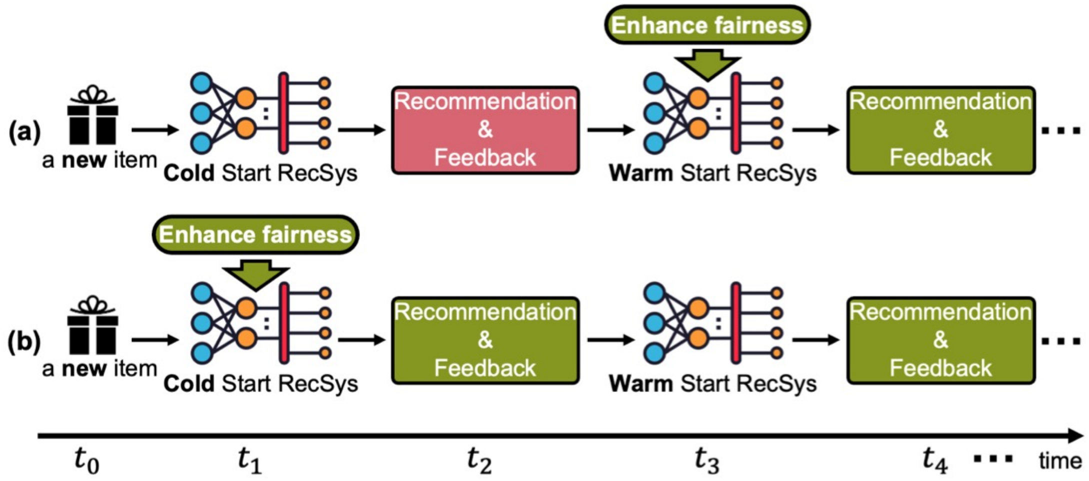
Diversifying Search Results using Self-Attention Network. Qin et al. CIKM 2020.

Fairness-aware Re-ranking

- Item fairness
  - Ensure each item/item group receives a fair proportion of exposure
  - Mitigating bias such as gender/politics, etc
- Neural fairness-aware methods are less explored
  - PLRank [Oosterhuis, 2021], FULTR [Yadav et al., 2021], GEN [Zhu et al., 2021]…

Fairness-aware Re-ranking: Fairness in Cold Start RS

- Motivation
  - The RS should treat different new items fairly in a cold-start scenario
  - Existing research focus on warm start scenario
- Solution
  - learnable post-processing framework



Fairness among New Items in Cold Start Recommender Systems. Zhu et al. SIGIR 2021.

Fairness-aware Re-ranking: Fairness in Cold Start RS

- Formalization of Fairness
  - Max-Min Opportunity Fairness
    - A model h^* is said to satisfy Max-Min Opportunity Fairness if it maximizes the true positive rate of the worst-off item

$$h^* = \arg\max_{h \in \mathcal{H}} \min_{i \in \mathcal{I}_c} TPR(i)$$

    - The true positive rate of an item

$$MDG_i = \frac{1}{|\mathcal{U}_i^+|} \sum_{u \in \mathcal{U}_i^+} \frac{\delta(\widehat{z}_{u,i} <= k)}{log(1 + \widehat{z}_{u,i})},$$

      - *MDG_i=0 means that item i is never recommended to matched users who like it during testing; MDG_i=1 means that i is ranked at the top position to all matched users during testing*

Fairness among New Items in Cold Start Recommender Systems. Zhu et al. SIGIR 2021.

Fairness-aware Re-ranking: Fairness in Cold Start RS

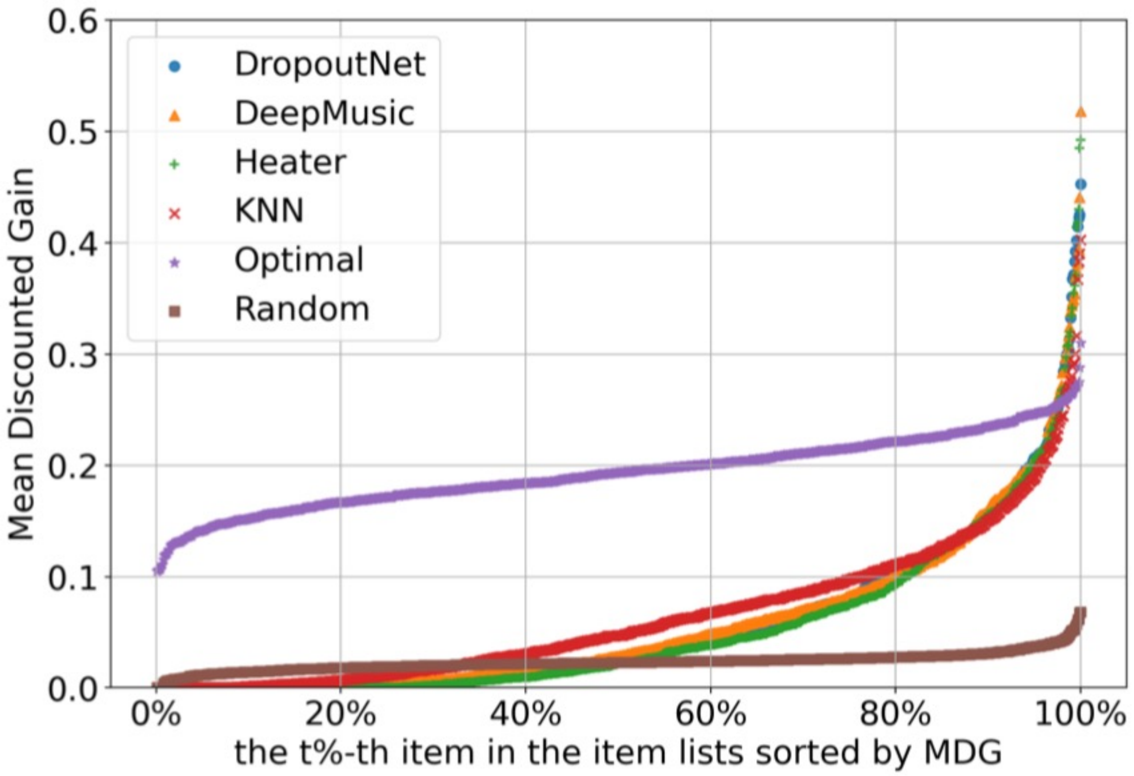
- Data-driven Study
  - Empirical results of four algorithms on ML1M

		Heater	DN	DM	KNN	Optimal	Random
User utility	NDCG@15	.5516	.5488	.5312	.4402	1.000	.0550
	NDCG@30	.5332	.5316	.5167	.4226	1.000	.0586
Item utility	MDG-all	.0525	.0552	.0572	.0646	.1932	.0236
Fairness	MDG-min10%	.0000	.0000	.0000	.0001	.1388	.0118
	MDG-min20%	.0000	.0000	.0001	.0020	.1498	.0145
	MDG-max10%	.2272	.2294	.2323	.2091	.2471	.0386

Fairness among New Items in Cold Start Recommender Systems. Zhu et al. SIGIR 2021.

Fairness-aware Re-ranking: Fairness in Cold Start RS

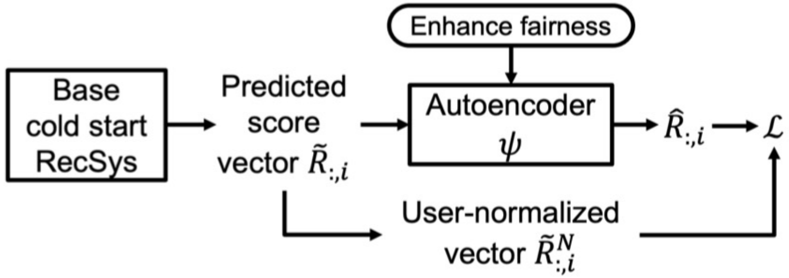
- Data-driven Study
  - MDG of items in ML1M



Fairness among New Items in Cold Start Recommender Systems. Zhu et al. SIGIR 2021.

Fairness-aware Re-ranking: Fairness in Cold Start RS

- Learnable Post-processing Framework
  - Requirement-1: promote under-served items so that their distributions of matched-user predicted scores
  - Requirement-2: for every user, the predicted scores follow the same distribution

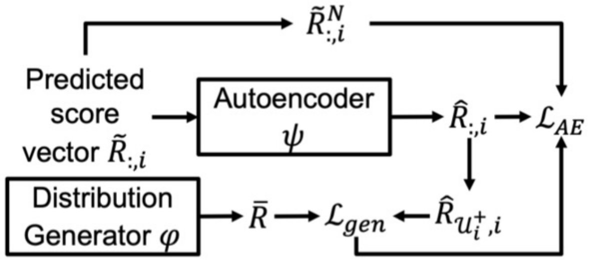


(a) The learnable post-processing framework.

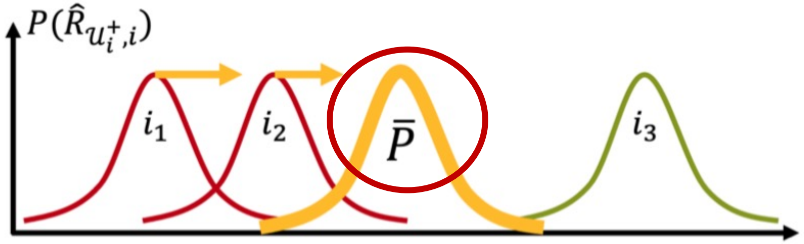
Fairness among New Items in Cold Start Recommender Systems. Zhu et al. SIGIR 2021.

Fairness-aware Re-ranking: Fairness in Cold Start RS

- The Joint-learning Generative Method
  - Framework & Intuition



(b) The Gen method.



  - 1st: Get the target distribution $\bar{P}$

$$\min_{\varphi} \mathcal{L}_{gen} = \sum_{i \in \mathcal{I}_w} MMD(\bar{R}, \hat{R}_{\mathcal{U}_i^+,i})$$

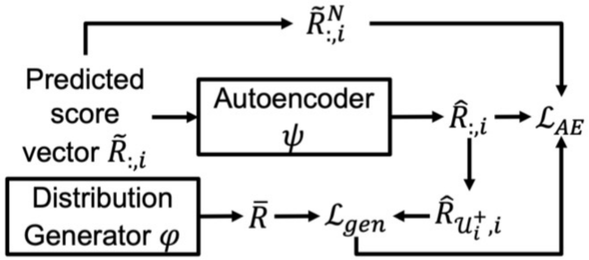
$$= \sum_{i \in \mathcal{I}_w} \Big(\frac{1}{S^2} \sum_{x,y=1}^{S} f(\bar{R}[x], \bar{R}[y]) - \frac{2}{S^2} \sum_{x=1}^{S} \sum_{y=1}^{S} f(\bar{R}[x], \hat{R}_{\mathcal{U}_i^+,i}[y])$$

$$+ \frac{1}{S^2} \sum_{x,y=1}^{S} f(\hat{R}_{\mathcal{U}_i^+,i}[x], \hat{R}_{\mathcal{U}_i^+,i}[y])\Big),$$

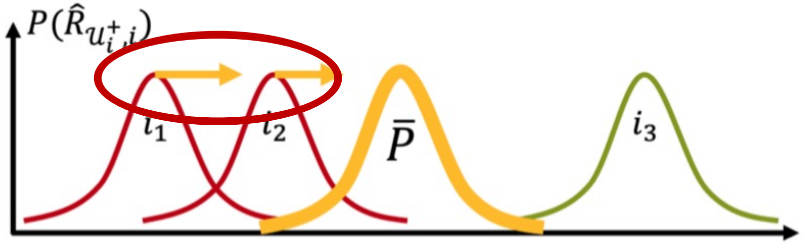
Fairness among New Items in Cold Start Recommender Systems. Zhu et al. SIGIR 2021.

Fairness-aware Re-ranking: Fairness in Cold Start RS

- The Joint-learning Generative Method
  - Framework & Intuition



(b) The Gen method.



  - 2nd: Update the autoencoder (the re-ranker)

$$\min_{\psi} \mathcal{L}_{AE} = \sum_{i \in \mathcal{I}_w} (\|\widetilde{R}^N_{:,i} - \widehat{R}_{:,i}\|_F + \alpha(MMD(\overline{R}, \widehat{R}_{\mathcal{U}_i^+,i}) \cdot \delta(i \in \mathcal{I}_{UE}))) + \lambda\|\psi\|_F,$$

Fairness among New Items in Cold Start Recommender Systems. Zhu et al. SIGIR 2021.

Fairness-aware Re-ranking: Fairness in Cold Start RS

- The Score Scaling Method
  - Intuition: up-scales the ratings of the unpopular items and down-scales ratings for popular items of high popularity

$$\widetilde{R}^{NS}_{\mathcal{U}_i^+,i} = \widetilde{R}^{N}_{\mathcal{U}_i^+,i} \times \frac{Max(\{Mean(\widetilde{R}^{N}_{\mathcal{U}_j^+,j})^{\beta} | j \in \mathcal{I}_w\})}{Mean(\widetilde{R}^{N}_{\mathcal{U}_i^+,i})^{\beta}},$$

  - Train the autoencoder (the re-ranker) by the loss

$$\min_{\psi} \mathcal{L}_{Scale} = \sum_{i \in \mathcal{I}_w} \|\widetilde{R}^{NS}_{:,i} - \widehat{R}_{:,i}\|_{\mathrm{F}} + \lambda \|\psi\|_{\mathrm{F}}.$$

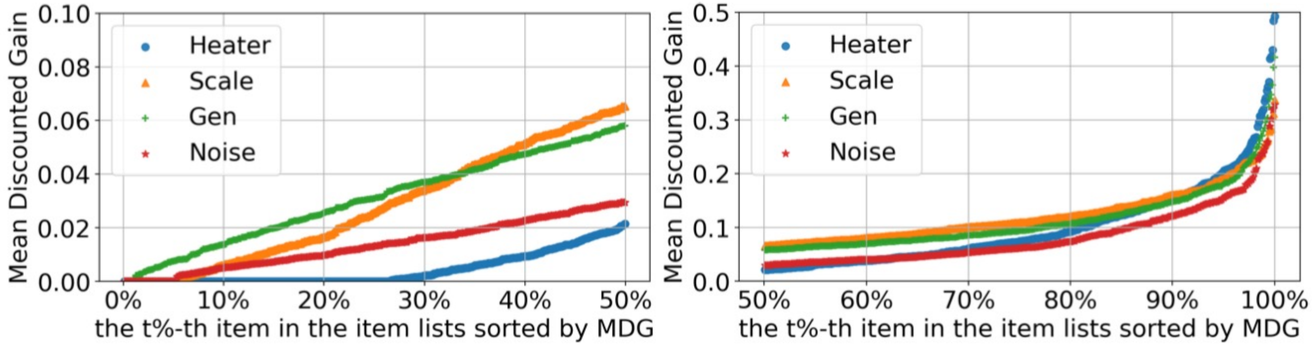
Fairness among New Items in Cold Start Recommender Systems. Zhu et al. SIGIR 2021.

Fairness-aware Re-ranking: Fairness in Cold Start RS

- Overall Performance

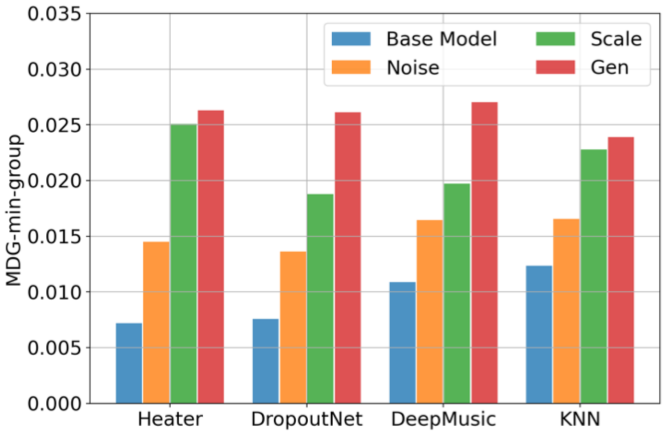
	NDCG @15	NDCG @30	MDG-all	Fairness: MDG min10%	Fairness: MDG min20%	Fairness: MDG max10%
Heater	0.5516	0.5332	0.0525	0.0000	0.0000	0.2272
Noise	0.4240	0.4084	0.0482	0.0017	0.0046	0.1730
Scale	0.5282	0.5135	0.0755	0.0015	0.0066	0.2025
Gen	0.5379	0.5206	0.0719	0.0073	0.0136	0.2036
DN	0.5488	0.5316	0.0552	0.0000	0.0000	0.2294
Noise	0.4586	0.4420	0.0513	0.0010	0.0037	0.1876
Scale	0.5315	0.5150	0.0766	0.0015	0.0069	0.2057
Gen	0.5345	0.5175	0.0745	0.0075	0.0138	0.2055
DM	0.5312	0.5167	0.0572	0.0000	0.0001	0.2323
Noise	0.4406	0.4304	0.0543	0.0007	0.0032	0.1937
Scale	0.5058	0.4946	0.0726	0.0010	0.0047	0.2140
Gen	0.5144	0.5024	0.0730	0.0027	0.0071	0.2136
KNN	0.4402	0.4226	0.0646	0.0001	0.0020	0.2091
Noise	0.3450	0.3378	0.0591	0.0016	0.0053	0.1643
Scale	0.4181	0.4027	0.0712	0.0023	0.0084	0.1791
Gen	0.4158	0.4002	0.0724	0.0075	0.0140	0.1831

- MDG of all Items



(a) 0~50%-th items. (b) 50~100%-th items.

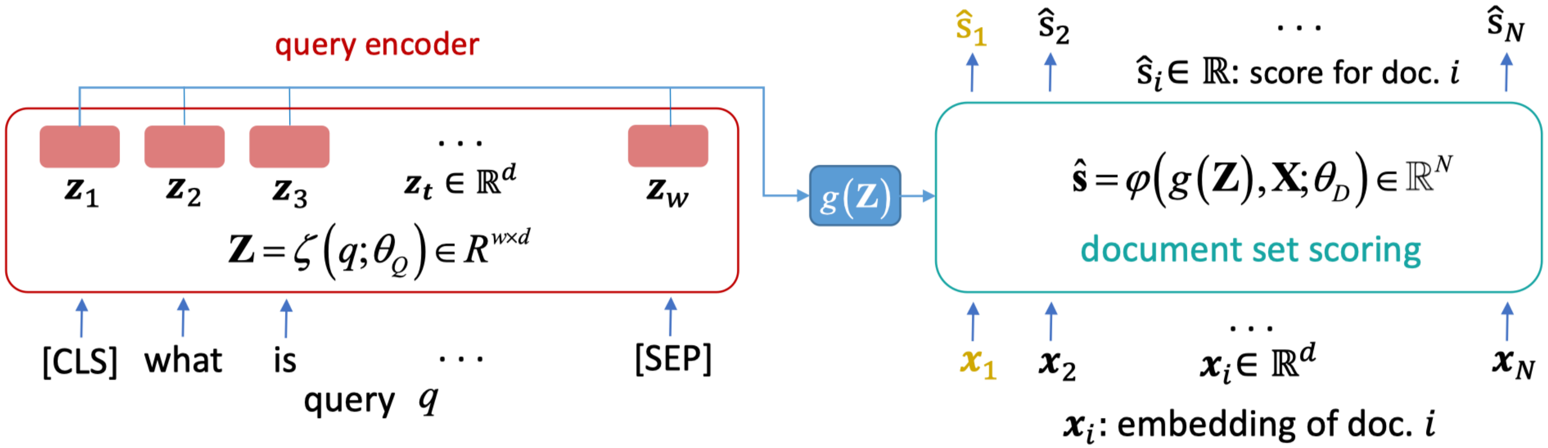
- Group-level fairness



Fairness among New Items in Cold Start Recommender Systems. Zhu et al. SIGIR 2021.

Fairness-aware Re-ranking: Mitigating Societal Bias

- Motivation
  - Mitigating societal bias in search results such as gender/politics biased results
- Structure



Mitigating Bias in Search Results Through Contextual Document Reranking and Neutrality Regularization. Zerveas et al.

Fairness-aware Re-ranking: Mitigating Societal Bias

- Training Objective
  - Relevance loss

$$\mathcal{L}_u(\mathbf{y}, \hat{\mathbf{s}}) = \mathrm{D_{KL}}(\sigma(\mathbf{y}) \,||\, \sigma(\hat{\mathbf{s}})) = -\sum_{i=1}^{N} \sigma(\mathbf{y})_i \log \frac{\sigma(\hat{\mathbf{s}})_i}{\sigma(\mathbf{y})_i}$$

  - Neutrality loss

$$\mathcal{L}_n(\mathbf{y}_n, \hat{\mathbf{s}}) = \mathrm{D_{KL}}(\sigma(\hat{\mathbf{s}}) \,||\, \sigma(\mathbf{y_n})) = -\sum_{i=1}^{C} \sigma(\hat{\mathbf{s}})_i \log \frac{\sigma(\mathbf{y_n})_i}{\sigma(\hat{\mathbf{s}})_i}$$

$$\mathcal{L}_{\mathrm{tot}} = \mathcal{L}_u + \lambda_r \mathcal{L}_n$$

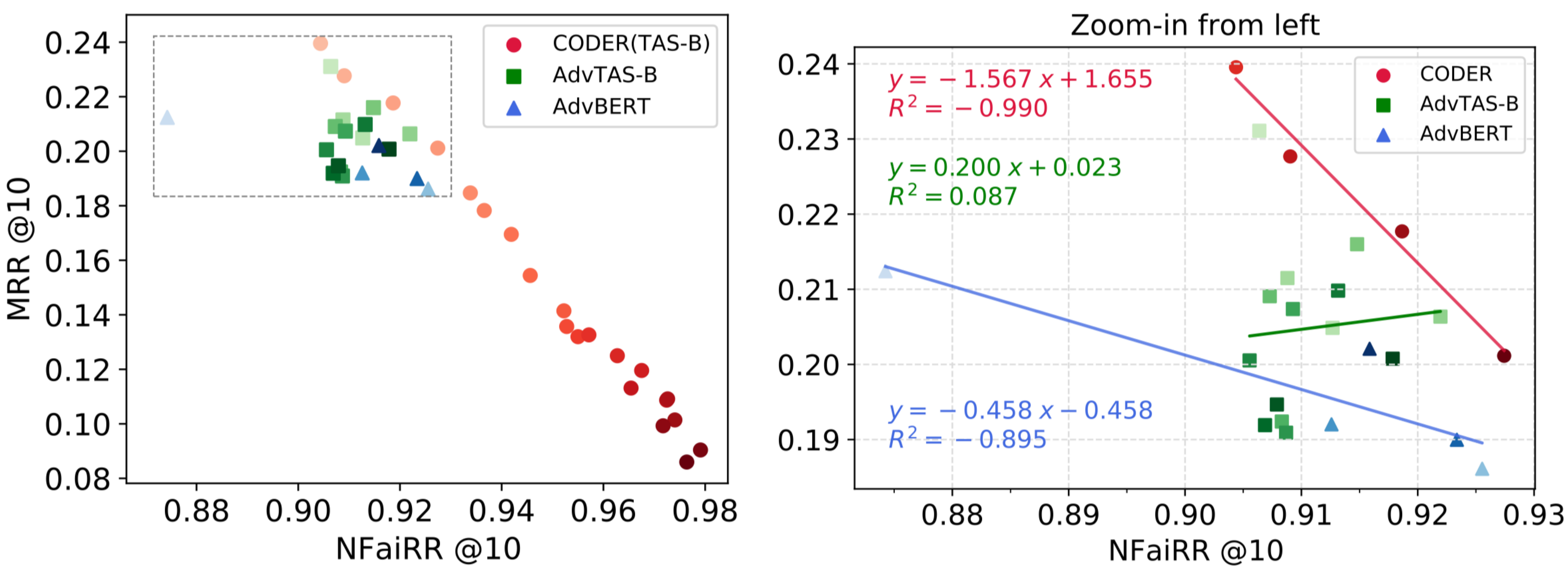
  - Neutrality score (based on pre-defined protected words such as gender)

$$mag^a(d) = \sum_{w \in \mathbb{V}_a} \#\langle w, d \rangle \qquad \omega(d) = \begin{cases} 1, & \text{if } \sum_{a \in A} mag^a(d) \leq \tau \\ 1 - \sum_{a \in A} \left| \frac{mag^a(d)}{\sum_{x \in A} mag^x(d)} - J_a \right|, & \text{otherwise} \end{cases}$$

Mitigating Bias in Search Results Through Contextual Document Reranking and Neutrality Regularization. Zerveas et al.

Fairness-aware Re-ranking: Mitigating Societal Bias

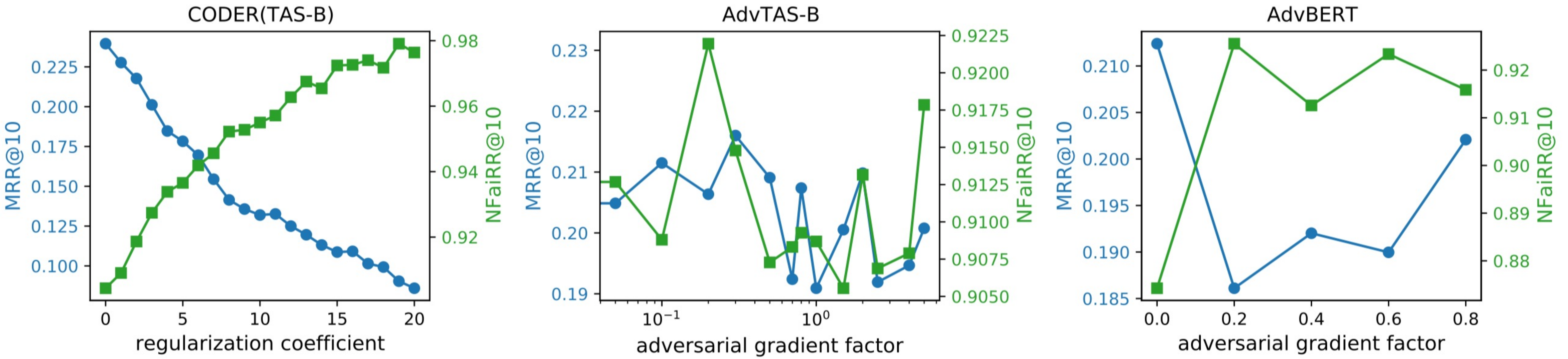
- Overall Performance



Mitigating Bias in Search Results Through Contextual Document Reranking and Neutrality Regularization. Zerveas et al.

Fairness-aware Re-ranking: Mitigating Societal Bias

- Controllable Regularization
  - The utility-fairness trade-off is more controllable and predictable



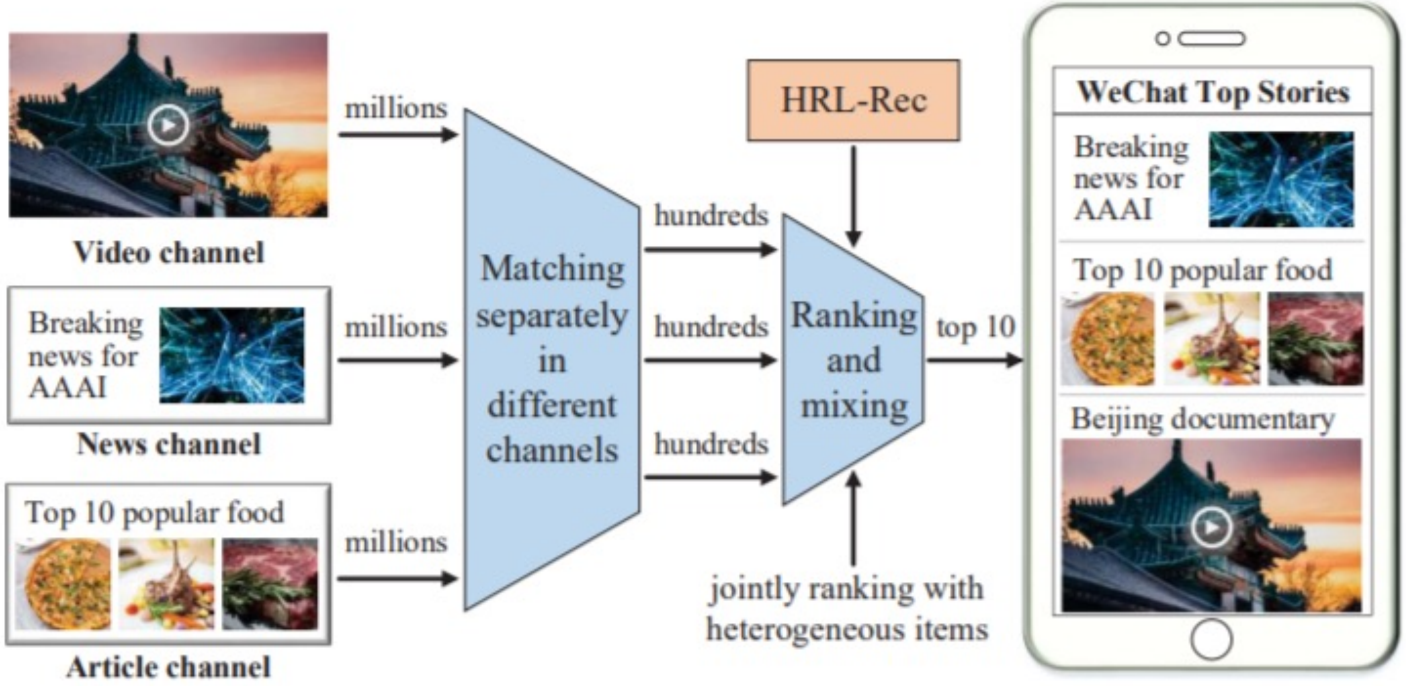
Mitigating Bias in Search Results Through Contextual Document Reranking and Neutrality Regularization. Zerveas et al.

Outline

- Introduction
  - Multi-stage recommender systems
  - Neural re-ranking
- Single objective: Accuracy oriented
  - Learning by observed signals
  - Learning by counterfactual signals
  - LibRerank library
- Multi-objective
  - Diversity-aware re-ranking
  - Fairness-aware re-ranking
- Emerging applications
- Summary

Emerging Applications: **Integrated Re-ranking**

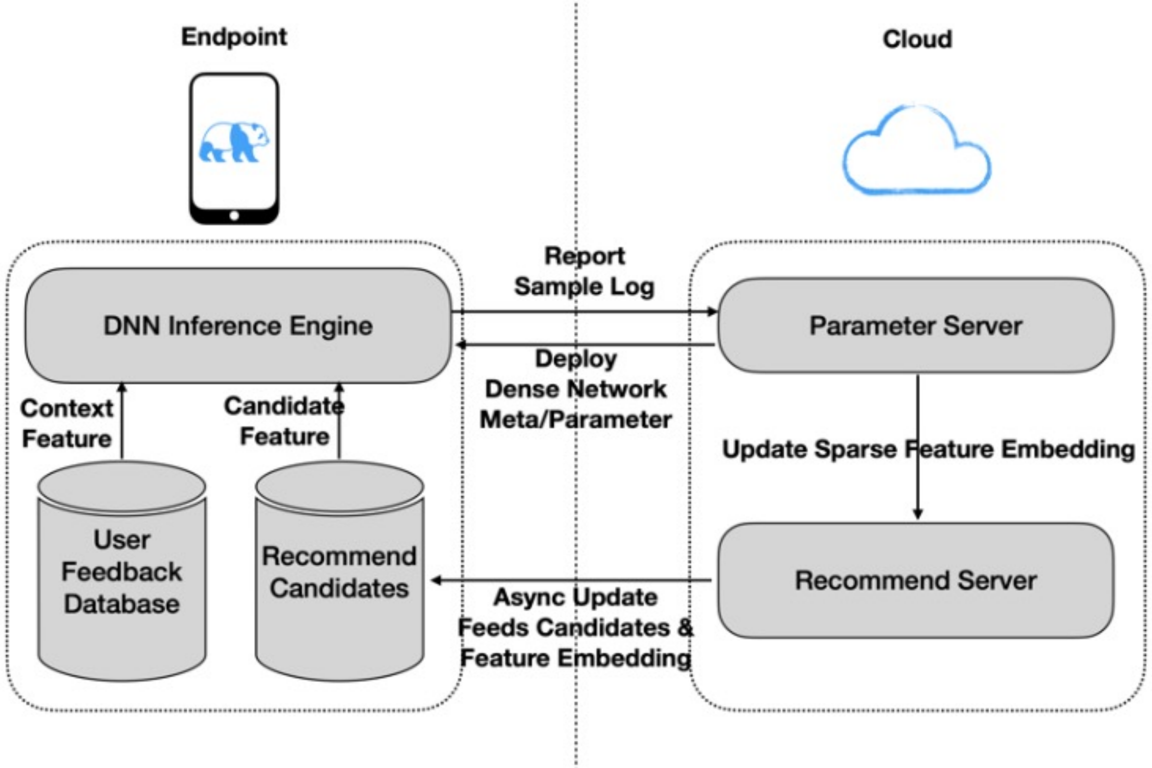
- Display **a mix of items** from sources with heterogeneous features
- The input is extended from a single list to **multiple lists**
- RL or hierarchical self-attention structure



Ruobing Xie, Shaoliang Zhang, Rui Wang, Feng Xia, and Leyu Lin. Hierarchical reinforcement learning for integrated recommendation. In AAAI, 2021.

Emerging Applications: **Edge Re-ranking**

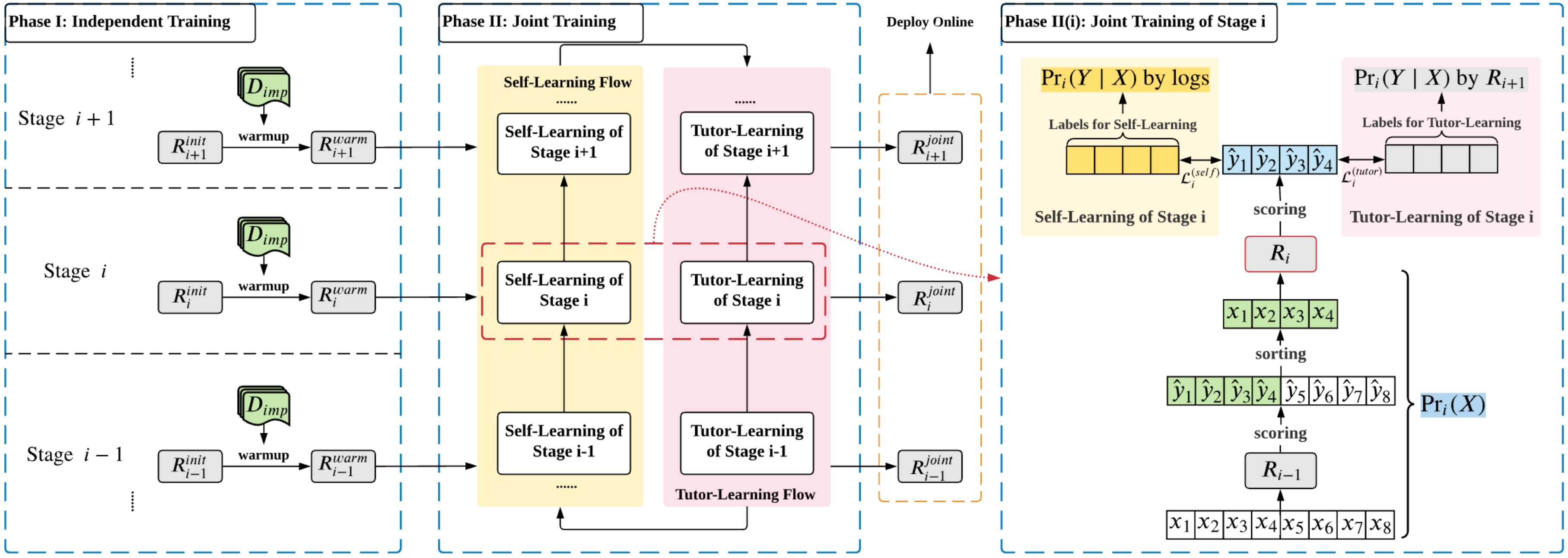
- EdgeRec: generates initial ranking lists on cloud, and conducts re-ranking with instant feedback on mobile devices



Yu Gong, Ziwen Jiang, Yufei Feng, et al. Edgerec: recommender system on edge in mobile taobao. In CIKM, 2020.

Emerging Applications: **Jointly Optimization**

- Joint Optimization of Multi-Stage Cascade Systems
  - Ranking ↔ Re-ranking

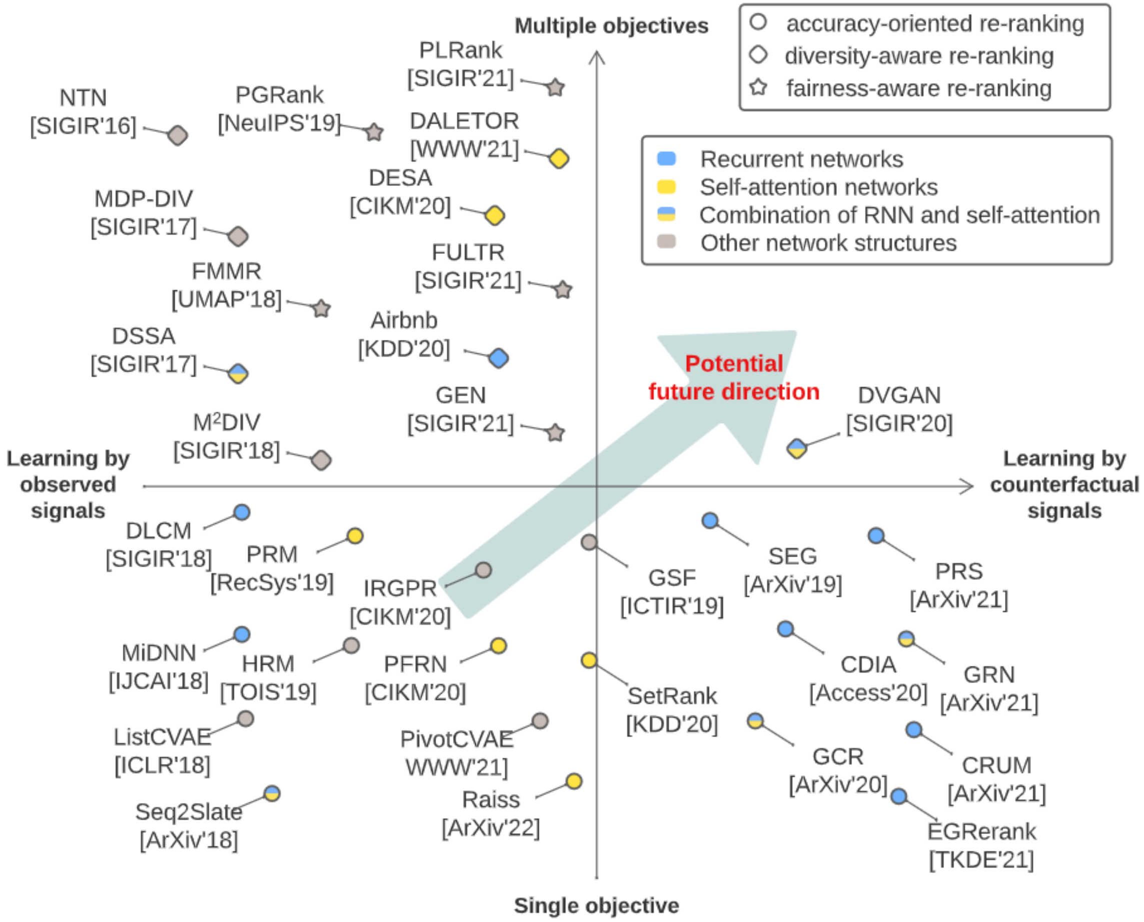


Jiarui Qin, Jiachen Zhu, Bo Chen, et al. RankFlow: Joint Optimization of Multi-Stage Cascade Ranking Systems as Flows. In CIKM, 2022.

Conclusion

Neural re-ranking has become a trending topic

- Learning by observed signals is **less noisy and easier to train**, but ignores other possible **permutations**
- Learning by counterfactual signals considers **all feasible permutations**, but the performance depends highly on the **quality** of the evaluator
- **Single objective** reranking model focus on the design of the listwise context modeling.
- **Multi-objective** reranking emphasize more on the balance between the objectives.



Future Directions

- Sparse feedback:
  - Only the feedback for the displayed lists can be observed
- Personalization for diversity/fairness:
  - Personalization is the core of recommender systems
  - Personalization for diversity/fairness remains less explored
- Tradeoff between multiple objectives:
  - Automatic balance between multiple objectives

- Diversity/fairness for integrated re-ranking:
  - Study the combined diversity effect for multiple channels
  - Explore the exposure fairness for multiple channels
- Model personalization and compression:
  - Each user has a personalized re-ranking model on the device
  - Edge models are required to be light weight and are low power consumption
- Joint training of multi-stage recommender systems:
  - Utilize information learned by other stages (e.g., parameter transfer, gradient transfer)

THANKS