Implicit Feedbacks are Not Always Favorable: Iterative Relabeled One-Class Collaborative Filtering against Noisy Interactions

Zitai Wang[1,2], Qianqian Xu[3*], Zhiyong Yang[1,2],
Xiaochun Cao[1,4], Qingming Huang[3,5,6,7*]

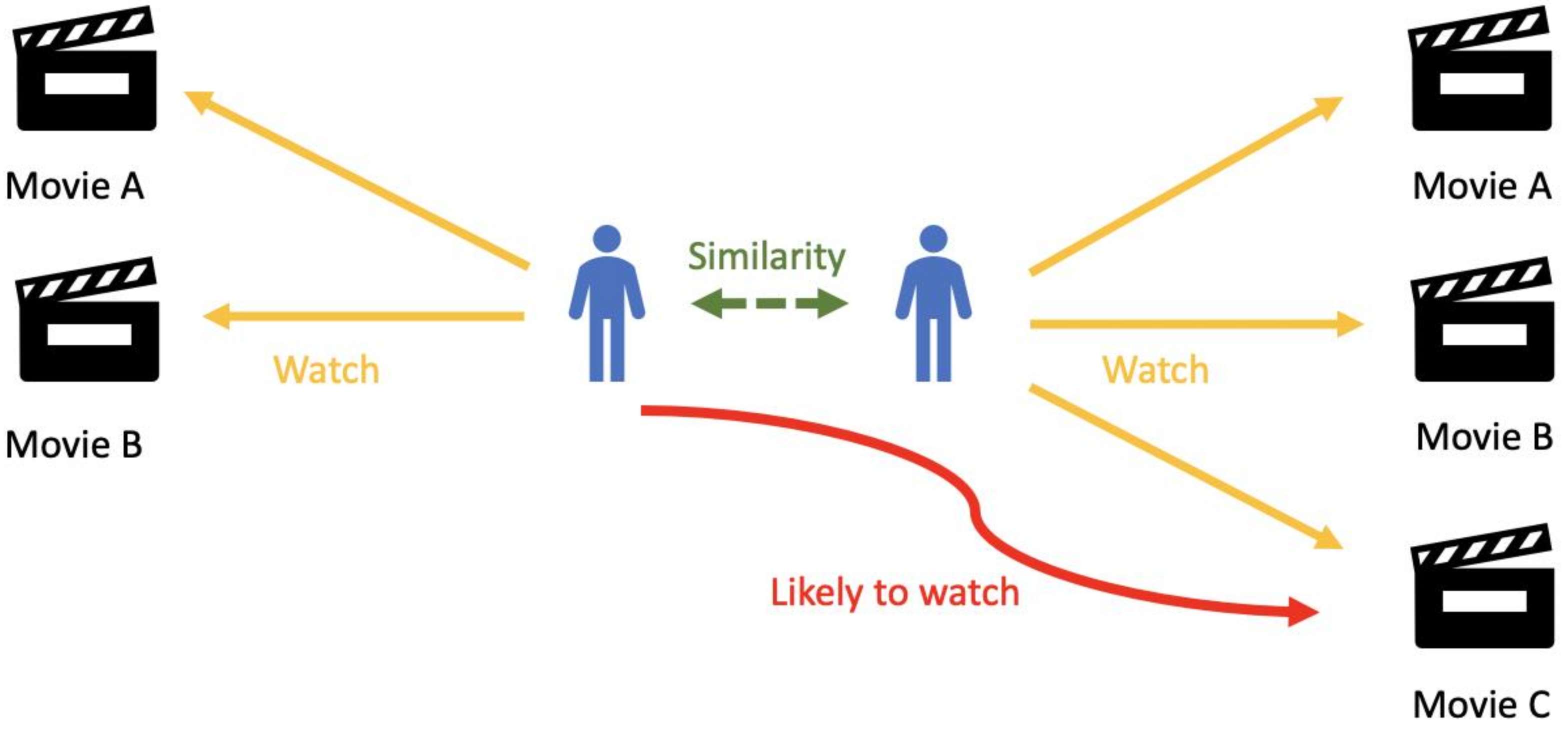
[1]SKLOIS, Institute of Information Engineering, CAS
[2]SCS, University of Chinese Academy of Sciences
[3]IIP, Institute of Computing Technology, CAS
[4]SCST, Sun Yat-sen University
[5]SCST, University of Chinese Academy of Sciences
[6]Key Lab. of BDKM, CAS
[7]Peng Cheng Laboratory

ACM multimedia
Chengdu, China OCT 20-24 2021

Outline

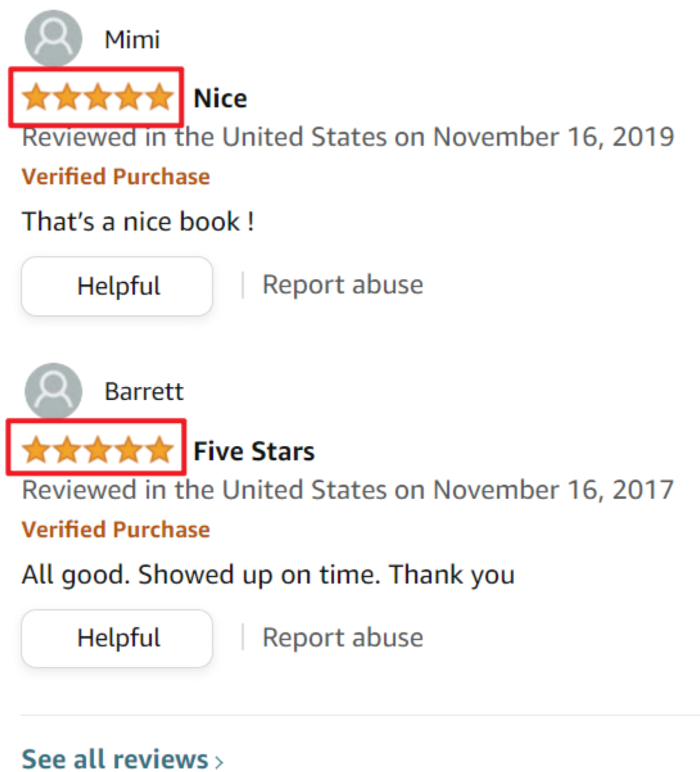
- Introduction
- Methodology
- Experiments
- Conclusion

Collaborative Filtering



Recommend according to users with similar feedback

Implicit Feedback



Explicit Feedback

Implicit Feedback

Noise in Implicit Feedback

This is a good movie, why no one told me before?

i_1	i_2	i_3	i_4
-	+	+	+
+	-	+	+
-	+	-	+
+	+	+	-

Ground-truth preference

i_1	i_2	i_3	i_4
0	1	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Interaction matrix, where red means noise.

i_1	i_2	i_3	i_4
0.2	0.8	0.1	0.8
0.9	0.3	0.8	0.9
0.9	0.9	0.3	0.9
0.9	0.9	0.8	0.3

Predicted scores

I search this movie for my friends, but I do not like it.

Prior arts

Reweighting
- WRMF, NA

Resampling
- DNS

Regularization
- iCD, SRRMF

	Pos Examples	"Neg" Examples
Uniform	$W_{ij} = 1$	$W_{ij} = \delta$
User-Oriented	$W_{ij} = 1$	$W_{ij} \propto \sum_j R_{ij}$
Item-Oriented	$W_{ij} = 1$	$W_{ij} \propto m - \sum_i R_{ij}$

Popular items are less likely to be negative

$$R_r = \frac{1}{2} \sum_{u} \sum_{i<j \notin E_u} (\hat{r}_{ui} - \hat{r}_{uj})^2$$

Unobserved items should have similar scores

Algorithm 1 Ranking-Aware Reject Sampling - Linear

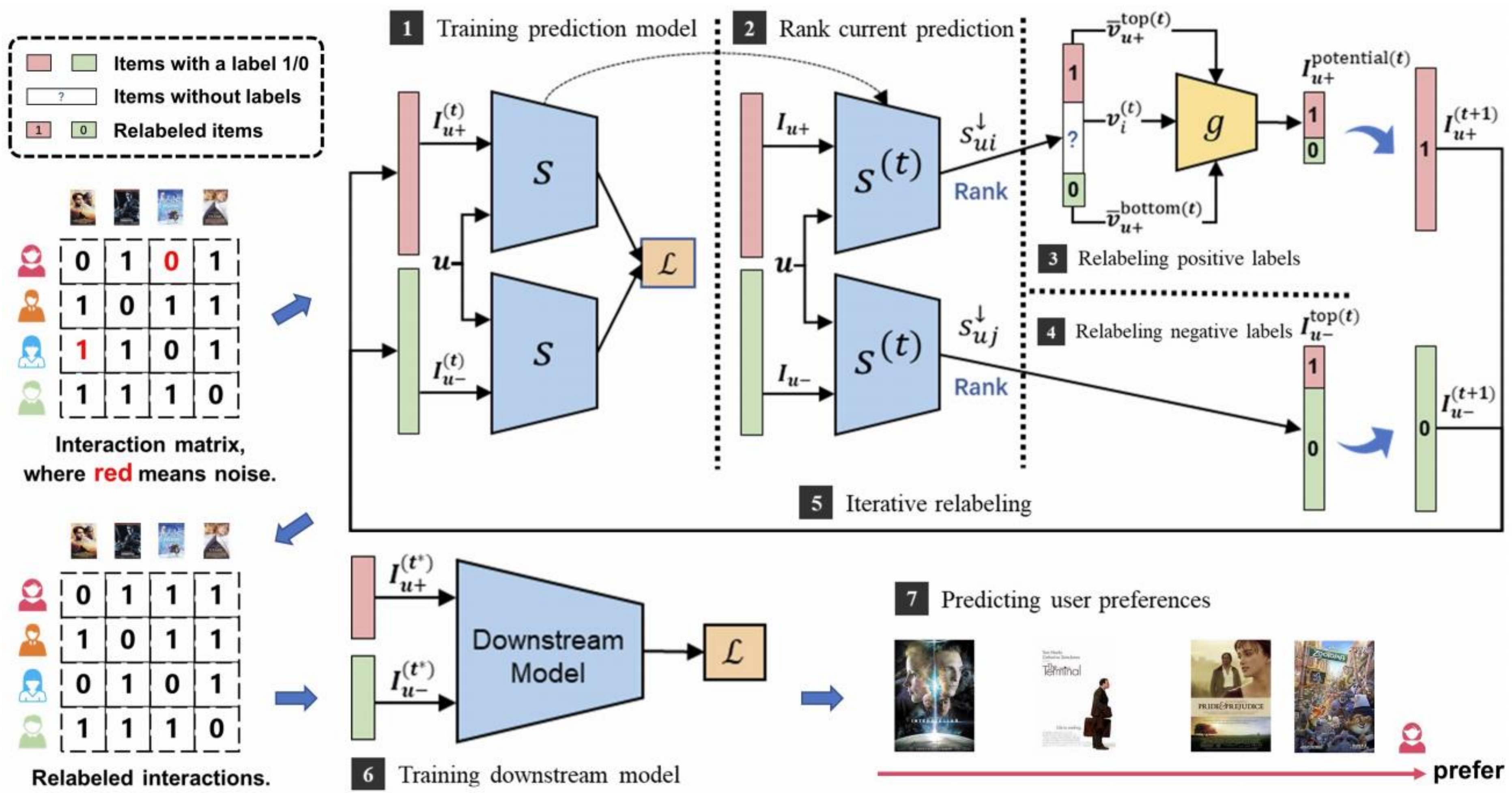
Require: Unobserved item set $I \backslash I_u$, scoring function $s(\cdot)$, parameter β
  Draw sample j, l uniformly from $I \backslash I_u$
  Query $s(j)$ and $s(l)$
  if $s(j) > s(l)$ **then**
    Return j with probability $\frac{1}{1+\beta}$, or return l otherwise
  else
    Return l with probability $\frac{1}{1+\beta}$, or return j otherwise
  end if

Items with lower scores are rejected

Outline

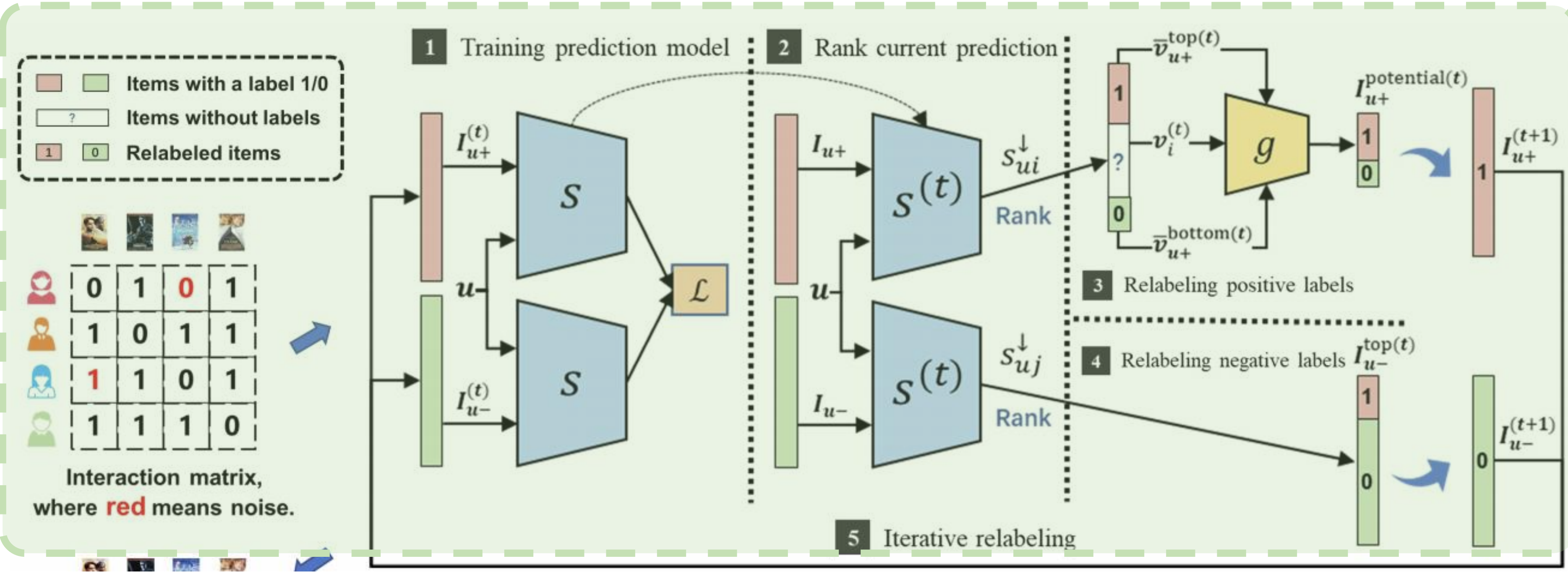
- Introduction
- Methodology
- Experiments
- Conclusion

Framework



Items with a label 1/0
Items without labels
Relabeled items
Interaction matrix, where red means noise.
Relabeled interactions.
1 Training prediction model
2 Rank current prediction
3 Relabeling positive labels
4 Relabeling negative labels
5 Iterative relabeling
6 Training downstream model
7 Predicting user preferences
Rank
Rank
Downstream Model
prefer

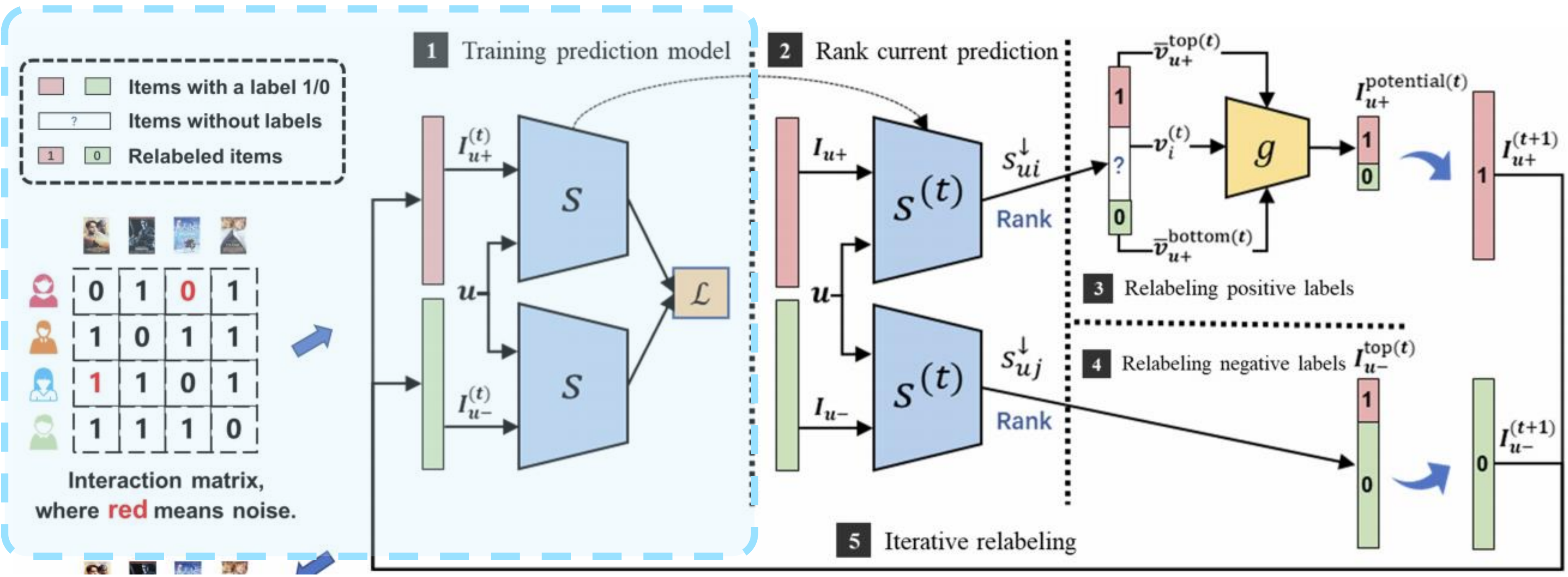
Framework



Main idea:

- Items with the top highest/lowest <u>scores</u> are more likely to be positive/negative
- Similar preference should have similar <u>embeddings</u>

Framework



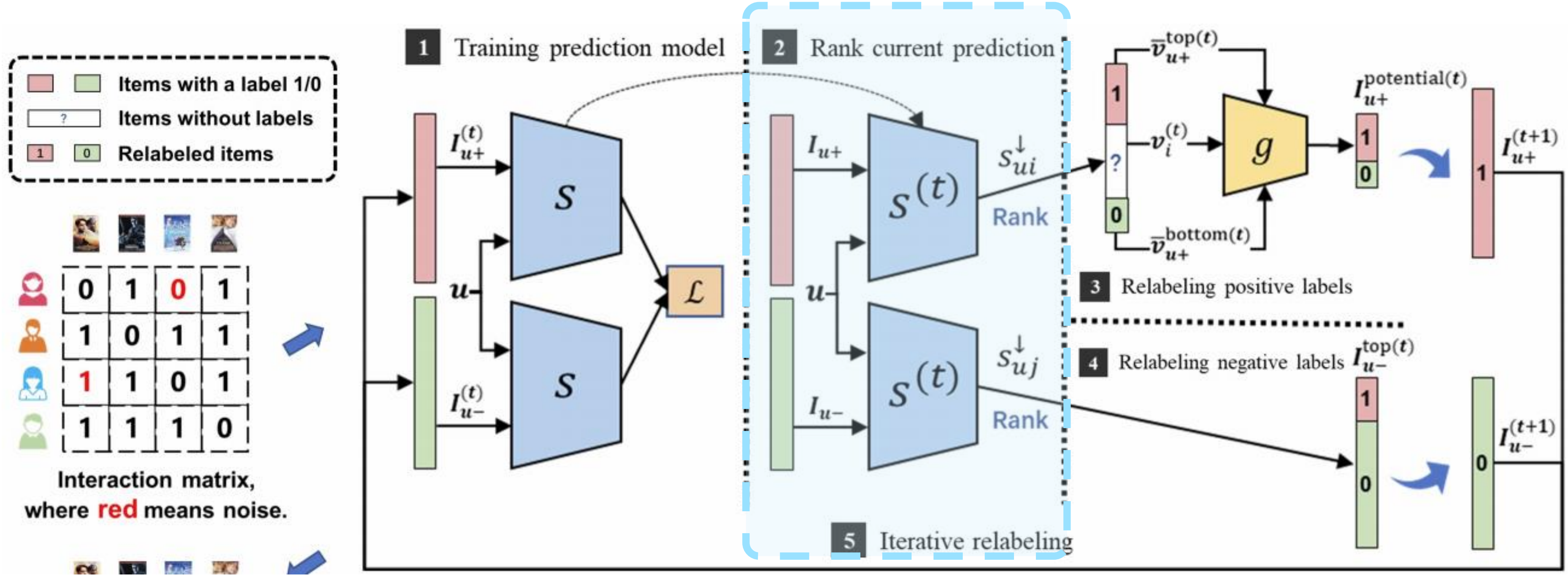
Train a prediction model based on the current label to evaluate the label confidence

$$\min_{\theta} - \sum_{(u,i,j)\in\mathcal{T}} \log \sigma \left(\boldsymbol{v}_u^T \boldsymbol{v}_i - \boldsymbol{v}_u^T \boldsymbol{v}_j \right)$$

$$= \{(u,i,j) | i \in \mathcal{I}_{u+}, j \in \mathcal{I}_{u-}\}$$

$$s_{u,i} = f(\boldsymbol{v}_u^*, \boldsymbol{v}_i^*) = \boldsymbol{v}_u^{*T} \boldsymbol{v}_i^* .$$

Framework



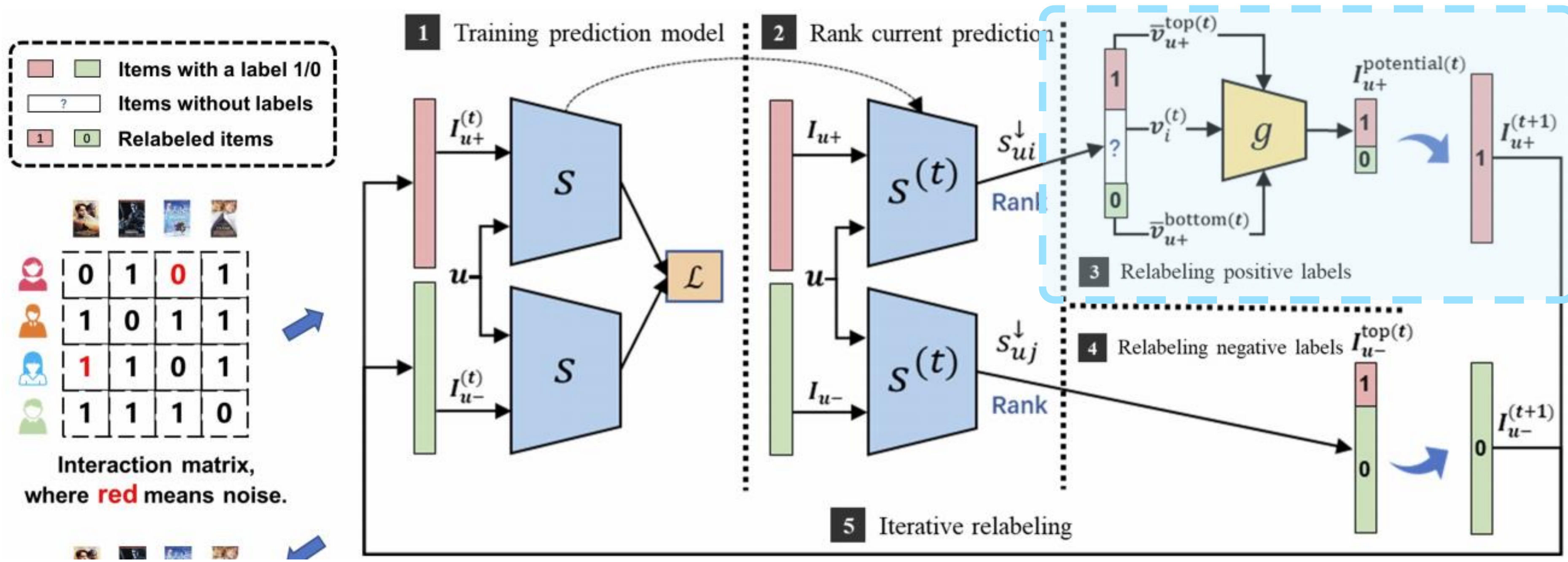
Rank current prediction, relabel the most confident samples

$$s^{\downarrow}_{u,1} > s^{\downarrow}_{u,2} > \cdots > s^{\downarrow}_{u,|\mathcal{I}_{u+}|}, u \in [n_{\text{user}}],$$

$$\mathcal{I}^{\text{top}}_{u+} = \left\{i : s_{u,i} \geq s^{\downarrow}_{u,n_+}\right\},$$

$$\mathcal{I}^{\text{bottom}}_{u+} = \left\{i : s_{u,i} \leq s^{\downarrow}_{u,n_-}\right\}$$

Framework



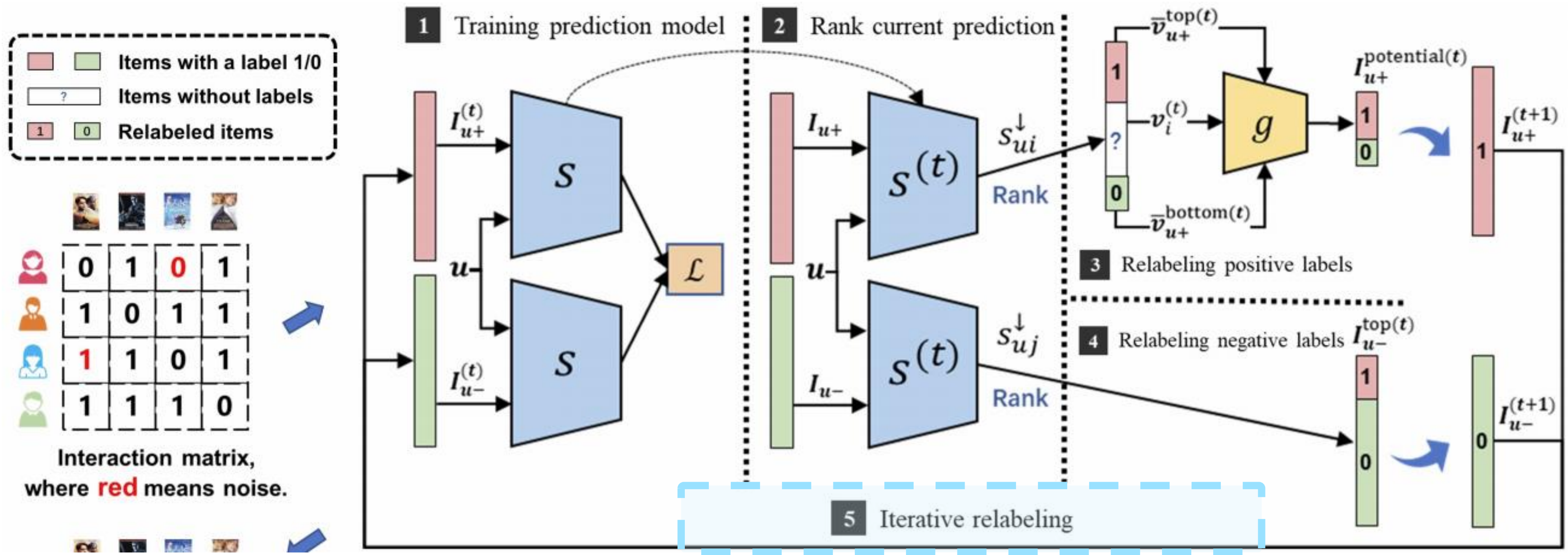
Relabel the rest samples according to embeddings

$$\overline{\boldsymbol{v}}_{u+}^{\text{top}} = \frac{1}{n_+} \sum_{i \in \mathcal{I}_{u+}^{\text{top}}} \boldsymbol{v}_i^*,$$

$$\overline{\boldsymbol{v}}_{u+}^{\text{bottom}} = \frac{1}{n_-} \sum_{i \in \mathcal{I}_{u+}^{\text{bottom}}} \boldsymbol{v}_i^*.$$

$$\hat{y}_{u,i} = g(\boldsymbol{v}_i^*) = \begin{cases} 1, & \alpha\|\boldsymbol{v}_i^* - \overline{\boldsymbol{v}}_{u+}^{\text{top}}\|_2 < \|\boldsymbol{v}_i^* - \overline{\boldsymbol{v}}_{u+}^{\text{bottom}}\|_2; \\ 0, & \text{otherwise,} \end{cases}$$

Framework



$$y_{u,i}^{(t)} = g_{\mu}(u,i) = \begin{cases} 1, & i \in \mathcal{I}_{u+}^{\text{top}(t)} \cup \mathcal{I}_{u+}^{\text{potential}(t)} \cup \mathcal{I}_{u-}^{\text{top}(t)}; \\ 0, & otherwise, \end{cases}$$

Outline

- Introduction
- Methodology
- Experiments
- Conclusion

Datasets

We perform evaluations on three real-world datasets

- To generate implicit training data, randomly select n_r ratings for each user as observed interactions, no matter whether the user likes the item or not.
- Binarize the rest ratings according to a threshold of 4 to evaluate the performance for user preference.

	MoiveLens100K	MoiveLens1M	Netflix
#User	943	6,040	8,143
#Item	1,683	3,706	17,770
#Rating	100,000	1,000,209	5,394,409
Sparsity	93.70%	95.53%	96.27%
Positive rate	56.43%	59.19%	54.79%

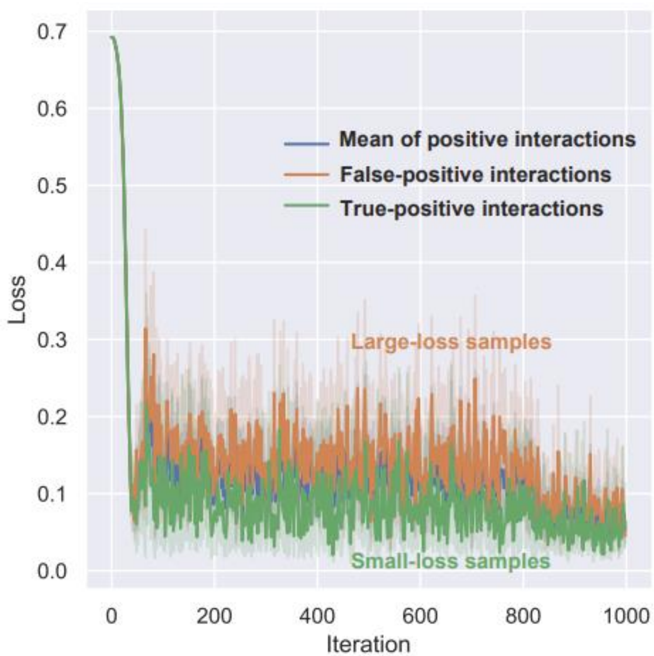
Competitors

Handle the noise in unobserved interactions:

- WRMF, NA, DNS, iCD, SRRMF

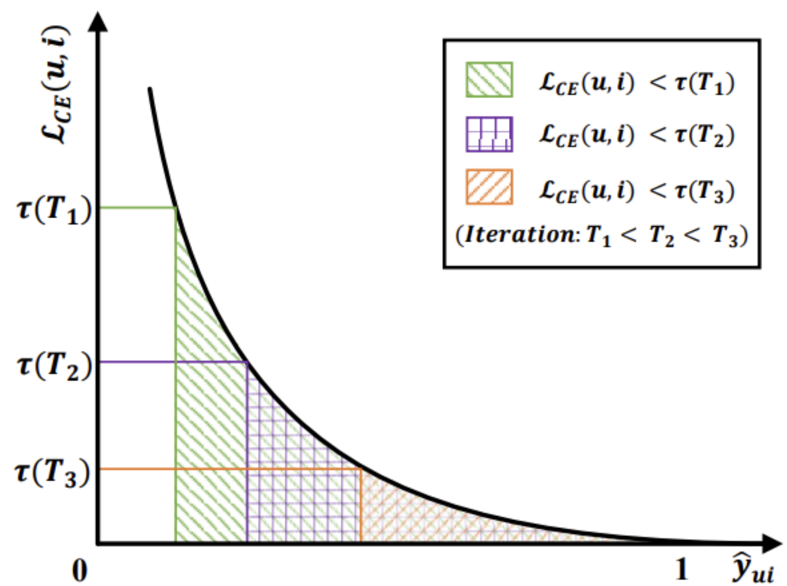
Handle the noise in observed interactions:

- T_CE



(b) Early training stages

$$\mathcal{L}_{T\text{-}CE}(u,i) = \begin{cases} 0, & \mathcal{L}_{CE}(u,i) > \tau \wedge \bar{y}_{ui} = 1 \\ \mathcal{L}_{CE}(u,i), & \text{otherwise,} \end{cases}$$



Results

From the results, we can find:

- The rate of positive samples increases gradually as the iterative relabeling process goes on
- Pseudo labels help improve the performance of the downstream model

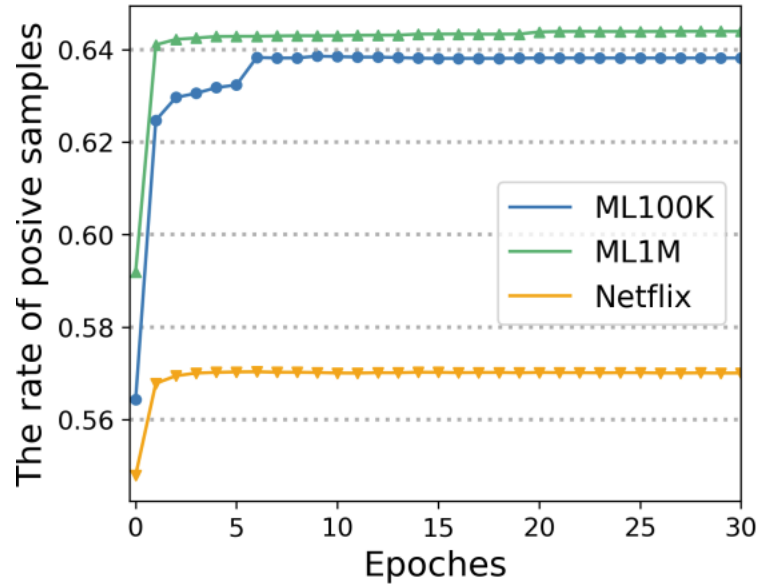


Table 2: Experimental results of our proposed method on three datasets. The downstream model is BPR.

Dataset	Relabeled	P@5	R@5	NDCG@5
ML100K	✓	**0.7481**	**0.1214**	**0.7635**
	×	0.7179	0.1178	0.7264
ML1M	✓	**0.7861**	**0.0979**	**0.7969**
	×	0.7762	0.0966	0.7791
Netflix	✓	**0.7607**	**0.1368**	**0.7754**
	×	0.7343	0.1315	0.7425

Table 3: Experimental results of our proposed method on three datasets. The downstream model is MLP.

Dataset	Relabeled	P@5	R@5	NDCG@5
ML100K	✓	**0.7513**	**0.1230**	**0.7727**
	×	0.7167	0.1195	0.7347
ML1M	✓	**0.7885**	**0.0977**	**0.7974**
	×	0.7782	0.0970	0.7870
Netflix	✓	**0.7301**	**0.1315**	**0.7448**
	×	0.7180	0.1295	0.7306

Results

ACM multimedia
Chengdu, China OCT 20-24 2021

Table 4: Experimental results on MovieLens100K.

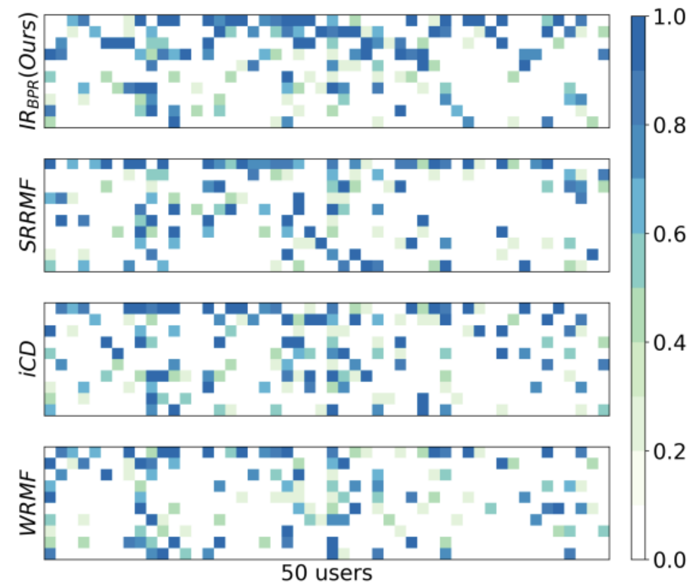
	P@1↑	P@5↑	P@10↑	R@5↑	R@10↑	NDCG@5↑	NDCG@10↑
DNS [44]	0.7606	0.7171	0.6825	0.1175	0.2201	0.7247	0.6980
NA [20]	0.7686	0.7111	0.6870	0.1190	0.2244	0.7225	0.7088
WRMF [25]	0.8048	0.7284	0.6855	0.1191	0.2187	0.7455	0.7099
iCD [2]	0.7706	0.7171	0.6982	0.1196	0.2245	0.7274	0.7104
SRRMF [4]	0.8129	0.7066	0.6730	0.1171	0.2195	0.7281	0.6967
T_CE [39]	0.7968	0.7272	0.6996	0.1212	0.2271	0.7413	0.7169
IR$_{BPR}$(Ours)	0.8229	0.7481	0.7040	0.1214	0.2272	0.7635	0.7269

Table 5: Experimental results one MovieLens1M.

	P@1↑	P@5↑	P@10↑	R@5↑	R@10↑	NDCG@5↑	NDCG@10↑
DNS [44]	0.7901	0.7773	0.7565	0.0960	0.1812	0.7805	0.7651
NA [20]	0.7846	0.7703	0.7548	0.0960	0.1804	0.7736	0.7619
WRMF [25]	0.8019	0.7843	0.7665	0.0971	0.1828	0.7893	0.7753
iCD [2]	0.8089	0.7838	0.7603	0.0953	0.1801	0.7897	0.7714
SRRMF [4]	0.8115	0.7809	0.7639	0.0969	0.1822	0.7933	0.7780
T_CE [39]	0.8167	0.7823	0.7612	0.0968	0.1820	0.7892	0.7723
IR$_{BPR}$(Ours)	0.8366	0.7861	0.7686	0.0979	0.1833	0.7969	0.7810

Table 6: Experimental results on Netflix.

	P@1↑	P@5↑	R@1↑	R@5↑	NDCG@5↑	NDCG@10↑
DNS [44]	0.7336	0.6935	0.0266	0.1246	0.7019	0.16874
NA [20]	0.7689	0.7187	0.0278	0.1288	0.7300	0.7086
WRMF [25]	0.7812	0.7185	0.0283	0.1295	0.7320	0.7057
iCD [2]	0.7980	0.7580	0.0290	0.1362	0.7672	0.7427
SRRMF [4]	0.7944	0.7549	0.0290	0.1358	0.7633	0.7399
T_CE [39]	0.7917	0.7501	0.0289	0.1349	0.7602	0.7310
IR$_{BPR}$(Ours)	0.8292	0.7607	0.0301	0.1368	0.7754	0.7433



Outline

- Introduction
- Methodology
- Experiments
- Conclusion

Conclusion

- We propose an iterative relabeling framework to eliminate the noise in both observed and unobserved interactions.

- The core of the framework lies the iterative relabeling module which aims to generate promising labels for user preferences by exploiting the self-training principle.

- The empirical experiments on three real-world datasets illustrate the effectiveness of our proposed framework。

THANKS!