子計畫名稱：子計畫二

子計畫主持人：吳尚鴻

展示技術：

多標籤攻擊防禦、即時鋼琴伴奏、Graph-based Recommendation

多標籤乾淨圖像後門攻擊的完整防禦方法

Yung-Cheng Chen, Zun-Lin Xu
Professor Shan-Hung Wu

Introduction

多標籤學習模型在圖像標註、物件偵測和文本分類等任務上皆有優異的性能表現。相較於單標籤模型，多標籤模型學習了標籤之間的關聯性，從而能達到更好的學習效果。因此，多標籤模型更適合應用在較為複雜的現實場域，如自駕車的即時辨識。

然而，近期出現的一種多標籤乾淨圖像後門攻擊利用了此種關聯性，透過修改部分標籤來植入後門，使得模型學習到錯誤的關聯，來達到攻擊的目的。

在本研究中，我們提出了一種能夠針對多標籤乾淨圖像後門攻擊的完整防禦方法，目的在減少受污染資料集的中毒率，也就是有毒資料在資料集中的占比。

具體來說，會分兩階段進行。第一階段的防禦稱為 potential poison masking，我們會先將被汙染的數據集進行粗略訓練，並依照每筆資料訓練的 loss進行排序，移除loss較大的部分資料集以減低資料集汙染程度。

第二階段的防禦稱為 data relabeling，為進一步提升防禦效果，我們利用前一階段的模型產出進行預測，並與原本受污染的資料及進行比對，對資料集重新標註，降低資料本身的中毒率，進而達成修復資料的目的。

Experiments

	COCO	VOC2012
train-set	82081	5717
trigger	人、車、交通號誌	人、車
poison rate	1.50%	4.80%
mAP (clean)	88.00	95.19

Result

COCO

	threshold	ASR(D)	ASR(A)	mAP
disappear				
backdoor	0.75	86.37%	X	88.90
defense	0.75	4.18%	X	87.22
appear				
backdoor	0.75	X	81.42%	89.21
defense	0.75	X	1.47%	79.54
misclassfication				
backdoor	0.75	82.42%	82.15%	88.86
defense	0.75	10.55%	9.78%	81.53

VOC2012

	threshold	ASR(D)	ASR(A)	mAP
disappear				
backdoor	0.75	90.52%	X	92.61
defense	0.75	2.59%	X	94.72
appear				
backdoor	0.75	X	83.41%	94.84
defense	0.75	X	0.00%	92.71
misclassfication				
backdoor	0.75	91.81%	98.69%	88.86
defense	0.75	5.17%	0.00%	81.53

Real-Time Piano Accompaniment

Kit Armstrong, Tzu-Ching Hung, Ji-Xuan Huang,
Professor Yi-Wen Liu

Introduction

To model the performance of classical music by human musicians, we explore collaborative music-making in a MIDI environment. In previous work, we presented a model that plays part of a score in real time together with a live musician playing the other part. We trained it to resemble human musicians faced with the same task, by tuning its systems built around a set of Kuramoto oscillators.

Accompaniment Model

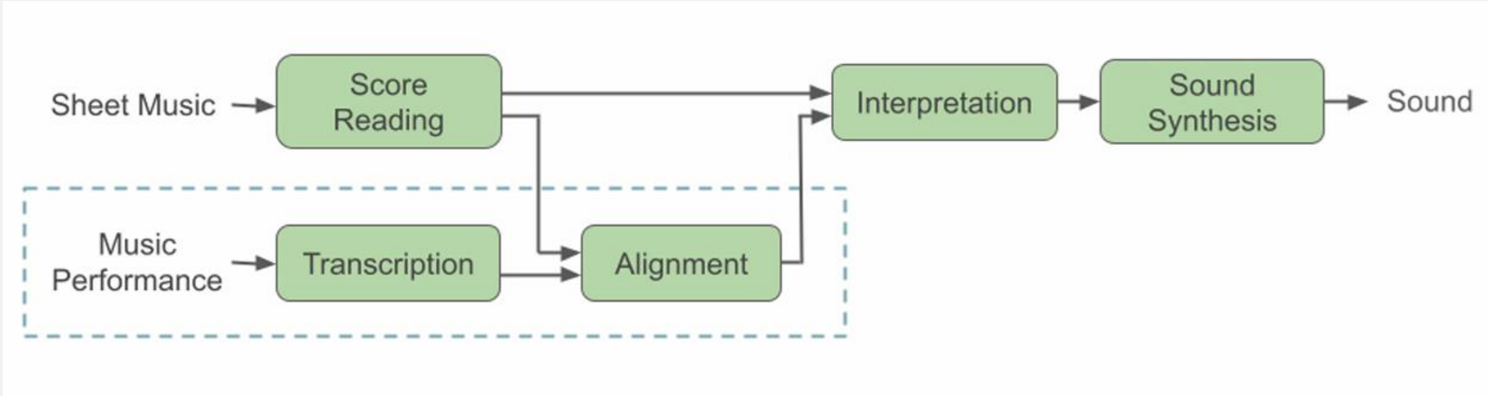


Fig. 1 The role of the Classical Musician

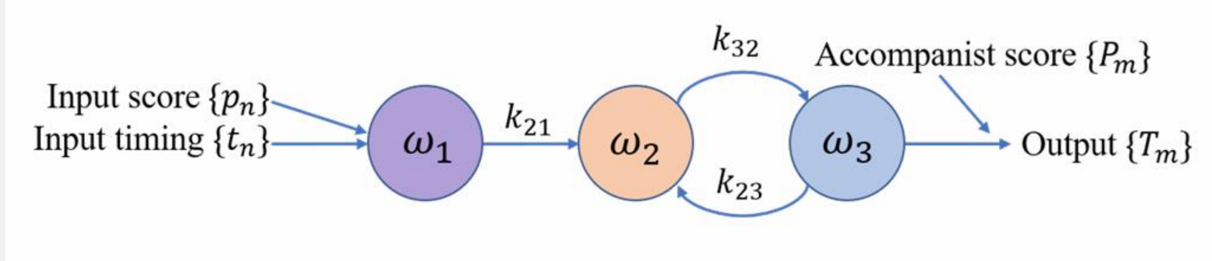
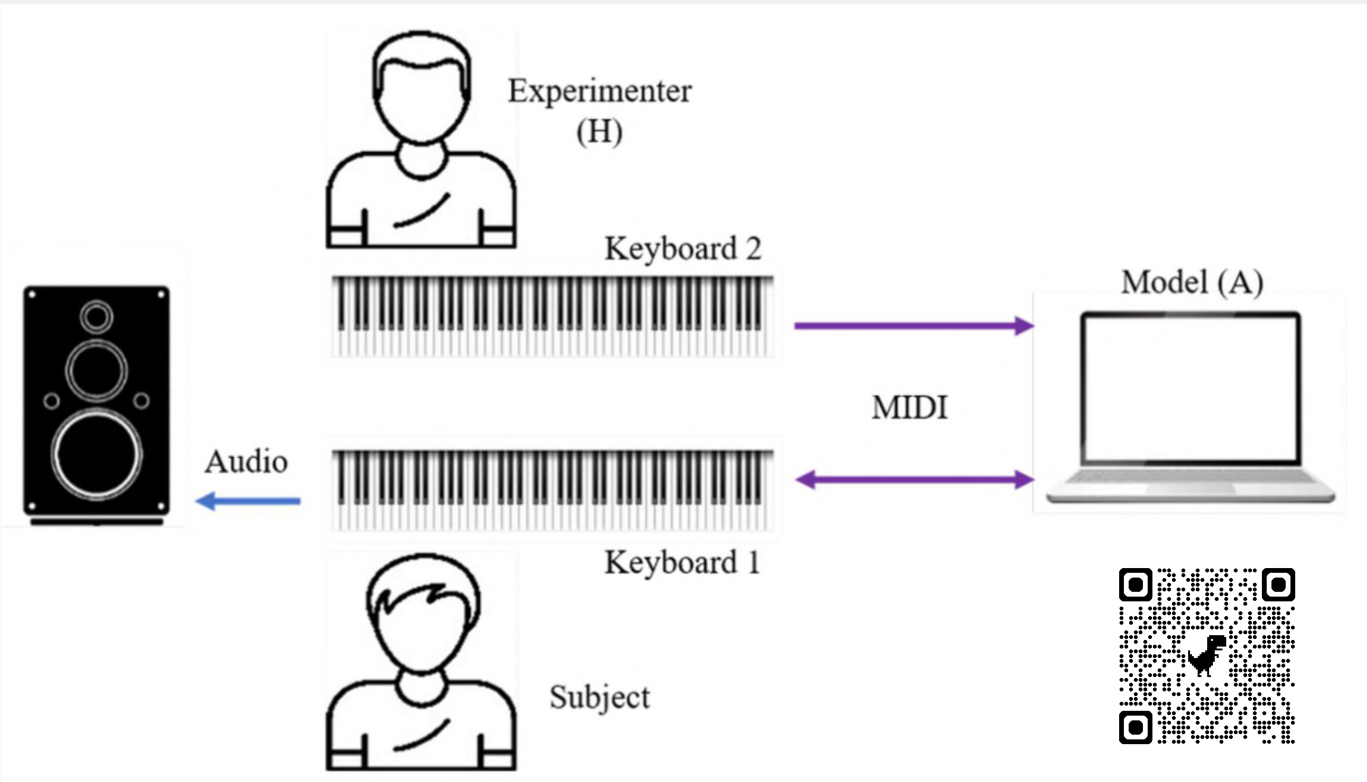


Fig. 2 Adapting Kuramoto Model

Here we chose 3 musical works and conducted experiments with a variety of pianists, recording the resulting performances as well as the testers' subjective impressions. We reconciled each performance with the corresponding music score, thereby defining a dataset which we call an "interpretation". In addition to subjective evaluation, we introduced objective criteria in the form of discriminants that classify interpretations as being the result of human-human interaction or of human-machine interaction. We considered the following qualities: desynchronization, jerkiness, and velocity curves. Our trained model performed similarly to humans with respect to the first two discriminants, but significantly differently with respect to the last. In light of this, it is notable that our experiment subjects often failed to correctly distinguish the two classes.

Experiments



Desynchronization

$$\Delta = \frac{1}{l}\sum_{k=1}^{l} |\tilde{s}_k - \tilde{U}_k|$$

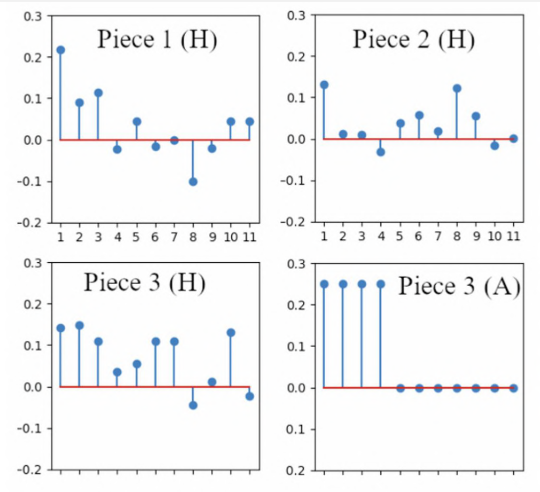
Jerk

$$J = \sum_{i=1}^{N-3} j_i^2 .$$

Piece	Trials	μ_J	σ_J
1 A	19	0.719×10^3	1.405×10^3
2 A	22	2.817×10^5	2.556×10^5
3 A	24	0.716×10^2	1.355×10^2
1 H	17	3.074×10^2	4.069×10^2
2 H	11	1.205×10^5	0.947×10^5
3 H	17	0.699×10^2	1.546×10^2

Statistics of total jerkiness in the A and the H trials

Velocity Curves



Regression coefficients for velocity prediction

https://smcnetwork.org/smc2024/papers/SMC2024_paper_id179.pdf

國立臺灣大學人工智慧研究中心
國立臺灣大學人工智慧技術暨全幅健康照護聯合研究中心
NTU JOINT RESEARCH CENTER FOR AI TECHNOLOGY AND ALL VISTA HEALTHCARE

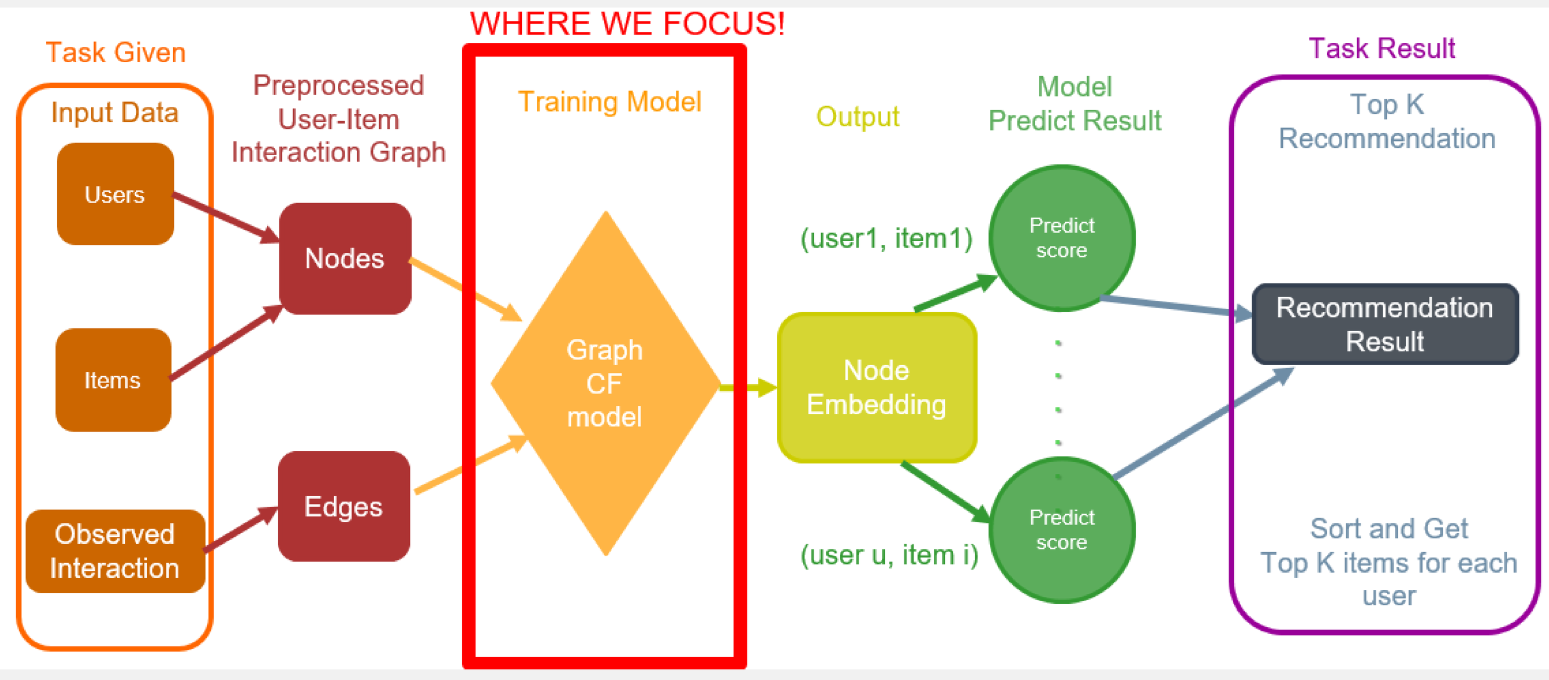
NSTC 國家科學及技術委員會
National Science and Technology Council

Improving Graph-based Recommendation with Unraveled Graph Learning

Chih-Chieh Chang[1], Diing-Ruey Tzeng[2], Chia-Hsun Lu[2], Ming-Yi Chang[3], Chih-Ya Shen[2*]
[1]The Management Undergraduate Program, National Taiwan University of Science and Technology, Taiwan.
[2*]Department of Computer Science, National Tsing Hua University, Taiwan.
[3]Department of Sociology, Fu Jen Catholic University, Taiwan.

Introduction

- Recommendation systems have been the subject of active study over the past few decades due to their importance and wide range of application scenarios.



Framework of Recommendation system

- Graph Collaborative Filtering (GraphCF) has achieved promising performance by leveraging the inferential power of Graph Neural Networks (GNNs).

- Here, we have two questions
  - **RQ1.** Does replacing graph augmentation with noise perturbation, as done in SimGCL, play a major role in enhancing performance?
  - **RQ2.** If the answer to the above question is NO, what are the key factors in enhancing performance?

PRELIMINARY ANALYSIS

- **RQ1**. Does replacing graph augmentation with noise perturbation, as done in SimGCL, play a major role in enhancing performance?

Table 1: Performance comparisons of the models that preserve/remove the ith embedding from the readout function. The suffixes $-V$, $-V_1$, and $-V_2$ indicate removing the 0th, 1st, and 2nd embedding, respectively. SimGCL by default does not include the 0th embedding in the readout function, and thus it is naturally the $-V$ version. The *underlined* values represent the best performance among various $-V_i$ versions for the same model, and the values in *bold* indicate the top-2 performance among all the compared approaches.

Dataset	Yelp-2018		Amazon-Book	
Model	Recall@20	NDCG@20	Recall@20	NDCG@20
LightGCN	0.0639	0.0525	0.0410	0.0318
SGL-ND	0.0644	0.0528	0.0440	0.0346
SGL-ND-V	0.0713	0.0594	0.0508	0.0402
SGL-ND-V_1	0.0573	0.0473	0.0416	0.0328
SGL-ND-V_2	0.0615	0.0512	0.0415	0.0327
SGL-ED	0.0675	0.0555	0.0478	0.0379
SGL-ED-V	0.0714	0.0597	0.0507	0.0403
SGL-ED-V_1	0.0612	0.0508	0.0416	0.0328
SGL-ED-V_2	0.0640	0.0540	0.0434	0.0341
SGL-RW	0.0667	0.0547	0.0457	0.0356
SGL-RW-V	**0.0720**	**0.0600**	**0.0524**	**0.0418**
SGL-RW-V_1	0.0619	0.0512	0.0419	0.0330
SGL-RW-V_2	0.0651	0.0543	0.0434	0.0342
SGL-WA	0.0671	0.0550	0.0466	0.0373
SGL-WA-V	0.0710	0.0594	0.0502	0.0402
SGL-WA-V_1	0.0652	0.0542	0.0430	0.0336
SGL-WA-V_2	0.0665	0.0554	0.0443	0.0347
SimGCL (default -V)	**0.0721**	**0.0601**	**0.0515**	**0.0414**
SimGCL-V_1	0.0652	0.0542	0.0515	0.0414
SimGCL-V_2	0.0666	0.0528	0.0515	0.0414

- **RQ2**. If the answer to the above question is NO, what are the key factors in enhancing performance?

$$E = \frac{1}{L+1}(\beta E^{(0)} + E^{(1)} + ... + E^{(L)}).$$

Table 2: Performance comparisons of models with different proportions of the 0th embedding in the readout function. SimGCL discards the 0th embedding by default, which is equivalent to setting $\beta = 0$.

Dataset	Yelp-2018		Amazon-Book	
Model	Recall@20	NDCG@20	Recall@20	NDCG@20
SimGCL ($\beta = 10$)	0.0335	0.0281	0.0179	0.0145
SimGCL ($\beta = 1$)	0.0591	0.0493	0.0376	0.0298
SimGCL ($\beta = 1e-1$)	0.0693	0.0580	**0.0460**	**0.0367**
SimGCL ($\beta = 1e-2$)	**0.0698**	**0.0583**	0.0457	0.0364
SimGCL ($\beta = 1e-3$)	0.0696	0.0582	0.0455	0.0361
SimGCL ($\beta = 1e-4$)	0.0696	0.0581	0.0455	0.0361
SimGCL ($\beta = 1e-5$)	0.0696	0.0581	0.0454	0.0360
SimGCL ($\beta = 0$)	*0.0689*	*0.0572*	*0.0453*	*0.0358*

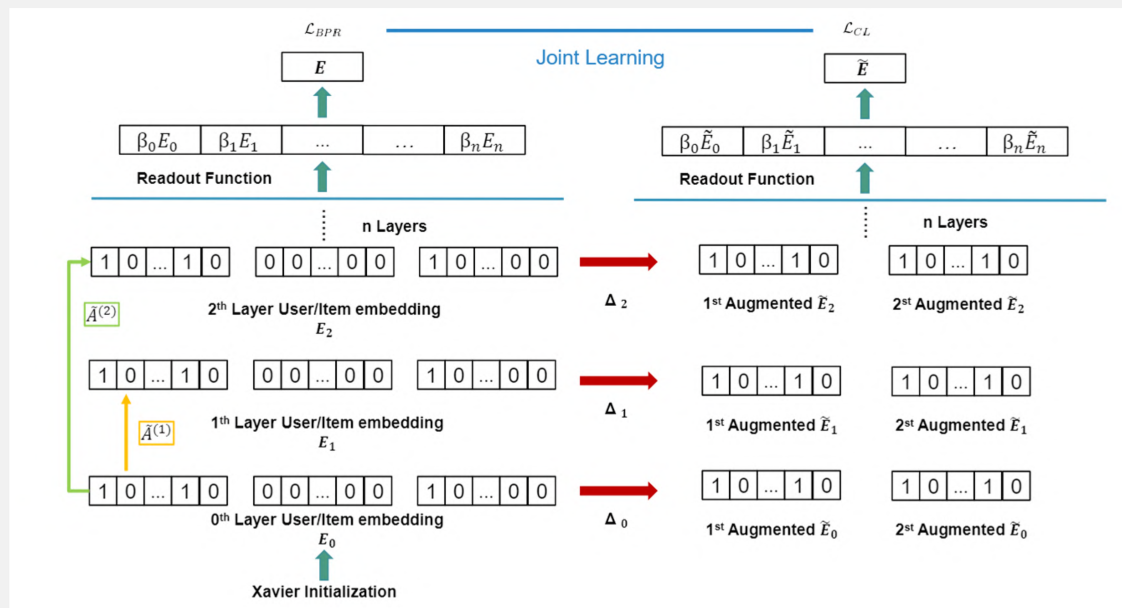
METHODOLOGY

- Unraveled Graph Contrastive learning (UGCL)
  - Finding the sweet spot of the embedding of each layer:

$$E = \frac{1}{L+1}(\beta_0 E^{(0)} + \beta_1 E^{(1)} + ... + \beta_L E^{(L)}),$$

  - Utilizing Contrastive learning, which can incorporate any strategy:

$$\tilde{E}^{(i)} = E^{(i)} + \Delta^{(i)}$$



Unraveled Graph Contrastive Learning (UGCL)

- Combine the teachers' output embeddings into a super teacher.
  - BPR Loss: $$\mathcal{L}_{BPR} = -\log\left(\sigma(e_u^T e_i - e_u^T e_j)\right),$$
  - CL Loss: $$\mathcal{L}_{CL} = \sum_{i \in B} -\log \frac{exp(z_i'^T z_i''/\tau)}{\sum_{j \in B} exp(exp(z_i'^T z_j''/\tau)},$$
  - Final Loss: $$\mathcal{L} = \mathcal{L}_{BPR} + \alpha \mathcal{L}_{CL}.$$

EXPERIMENTAL RESULTS

- Dataset:

Dataset	Users	Items	Interactions	Sparsity
Douban-Book	13,024	22,347	792,062	0.00272
Yelp-2018	31,668	38,048	1,561,406	0.00130
Amazon-Book	52,643	91,599	2,984,108	0.00062
ML-1M	6,040	3,492	575,281	0.02728

- Top *K* recommendation:

	Dataset	Douban-book		Yelp-2018		Amazon-book		ml-1M	
	Model	Recall@20	NDCG@20	Recall@20	NDCG@20	Recall@20	NDCG@20	Recall@20	NDCG@20
GraphCF	LightGCN	0.1501	0.1282	0.0639	0.0525	0.0411	0.0318	0.2618	0.3032
Graph Contrastive CF	SGL-ND	0.1626	0.1450	0.0658	0.0538	0.044	0.0346	0.2547	0.2972
	SGL-ED	0.1732	0.1551	0.0675	0.0555	0.0478	0.0379	0.2665	0.3200
	SGL-RW	0.1730	0.1546	0.0667	0.0547	0.0457	0.0356	0.2645	0.3091
	SGL-WA	0.1705	0.1525	0.0671	0.0550	0.0466	0.0373	0.2281	0.2721
	NCL	0.1483	0.1217	0.0611	0.0503	0.0400	0.0370	0.2603	0.2965
	SimGCL	0.1772	0.1583	0.0721	0.0601	0.0515	0.0414	0.2785	0.3240
SSL Rec	Multi-VAE	0.1310	0.1103	0.0584	0.0450	0.0407	0.0315	0.2766	0.3001
	SSL4Rec	0.1360	0.1148	0.0483	0.0382	0.0438	0.0337	0.0617	0.0557
	BUIR	0.1127	0.0893	0.0487	0.0404	0.0260	0.0209	0.1978	0.2425
New Sampling Strategy	MixGCF	0.1731	0.1552	0.0713	0.0589	0.0485	0.0378	0.1982	0.2131
	UGCL	**0.1825 (+22.9%)**	**0.1688 (+32.7%)**	**0.0727 (+16.9%)**	**0.0608 (+20.6%)**	**0.0549 (+33.6%)**	**0.0441 (+40%)**	**0.2841 (+8.5%)**	**0.3312 (+9.2%)**

- Performance comparisons of non-graph-based method:

Table 5: Performance comparisons of non-graph-based method with the proposed UGCL.

Dataset	Douban-Book		Yelp-2018		Amazon-Book		ML-1M	
Model	Recall@20	NDCG@20	Recall@20	NDCG@20	Recall@20	NDCG@20	Recall@20	NDCG@20
BPRMF (Rendle et al, 2012)	0.1302	0.1072	0.0501	0.0412	0.0303	0.02331	0.2442	0.2810
NeuMF (He et al, 2017)	0.1284	0.1062	0.0489	0.0403	0.0283	0.0211	0.2253	0.2639
Multi-VAE (Liang et al, 2018)	0.1310	0.1103	0.0584	0.0450	0.0407	0.0315	0.2766	0.3001
UGCL (Ours)	**0.1825**	**0.1688**	**0.0727**	**0.0608**	**0.0549**	**0.0441**	**0.2841**	**0.3312**

- Fairness on recommendation:

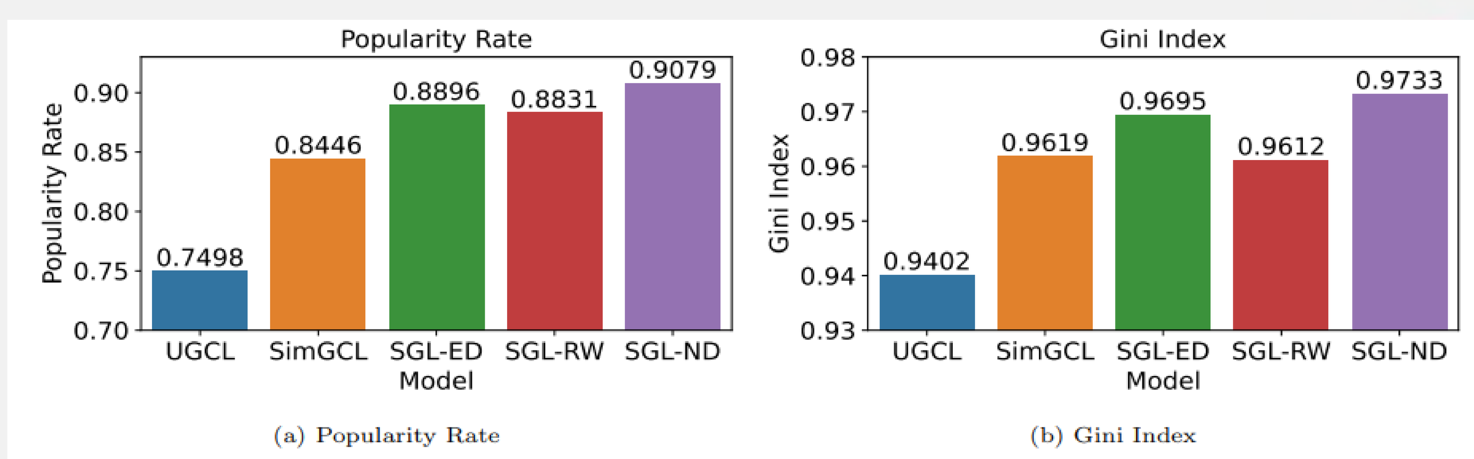


Fig. 4: Comparisons of Popularity Rate and Gini Index. A smaller value indicates a fairer recommendation.