The 17th ACM Conference on Recommender Systems
Singapore, SG, 09-20-2023
Main Track - Reproducibility

Challenging the Myth of Graph Collaborative Filtering: a Reasoned and Reproducibility-driven Analysis

Vito Walter Anelli[1], **Daniele Malitesta[1]**, Claudio Pomo[1], Alejandro Bellogín[2], Eugenio Di Sciascio[1], Tommaso Di Noia[1]

[1] *Politecnico di Bari, Bari (Italy),* ***email: firstname.lastname@poliba.it***
[2] *Universidad Autónoma de Madrid, Madrid (Spain),* ***email: alejandro.bellogin@uam.es***

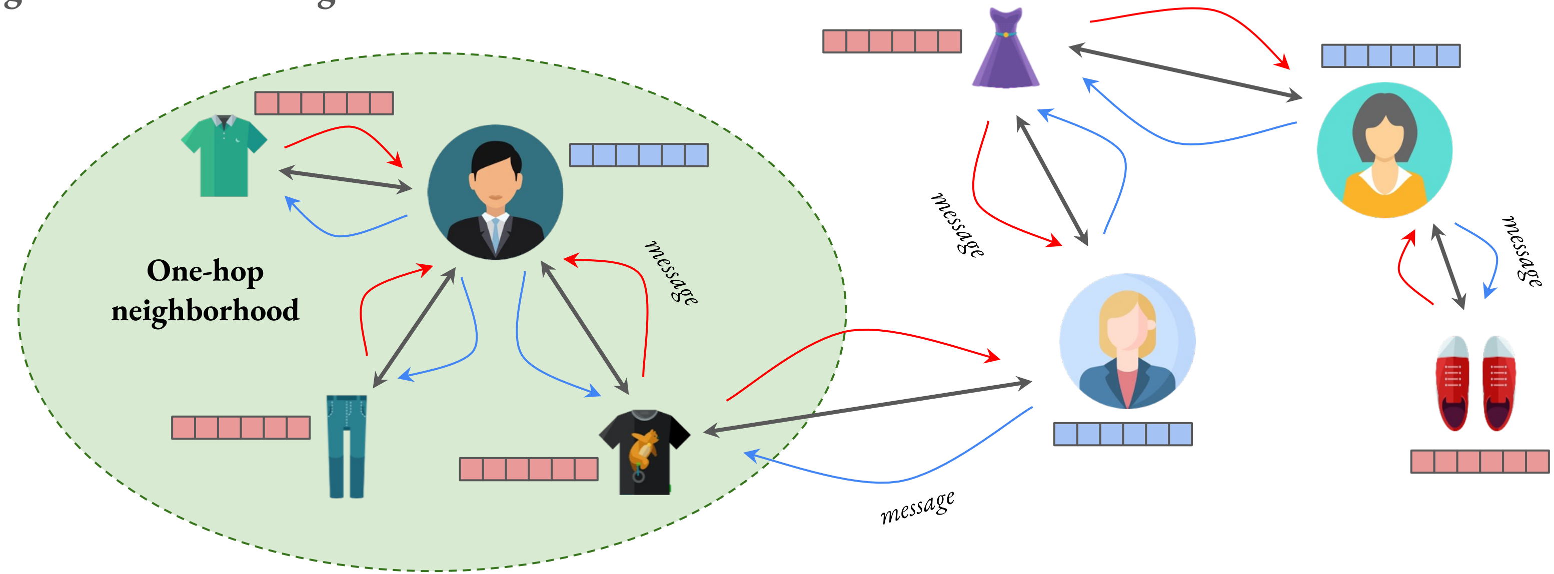
Outline

- Introduction and motivations
- Background and reproducibility analysis
- Replication of prior results (RQ1)
- Benchmarking graph CF approaches using alternative baselines (RQ2)
- Extending the experimental comparison to new datasets (RQ3 - RQ4)
- Conclusion and future work

Introduction and motivations

Graph collaborative filtering: message-passing

In **collaborative filtering** (CF), graph convolutional networks (**GCNs**) have **gained momentum** thanks to their ability to **aggregate neighbor** nodes **information** into ego nodes at multiple hops (i.e., **message-passing**), thus effectively **distilling** the **collaborative signal**



Graph collaborative filtering: a non-exhaustive timeline

2017-2018

Pioneer approaches proposing **GCN-based** aggregation methods **[van den Berg et al., Ying et al.]**

2019

Explore **inter-dependencies** between **nodes** and their **neighbors** **[Wang et al. (2019a)]**, use **graph attention networks** to recognize **meaningful** user-item **interactions** at **higher-grained** level **[Wang et al. (2019b)]**

2020

Lighten the **graph convolutional** layer **[Chen et al., He et al.]**, use **graph attention networks** to recognize **meaningful** user-item **interactions** at **higher-grained** level **[Wang et al. (2020), Tao et al.]**

2021-2022

Self-supervised and **contrastive learning** **[Wu et al., Yu et al.]**

2021-2022

Simplify the **message-passing** formulation **[Mao et al., Peng et al., Shen et al.]**

2021-2022

Explore **other latent spaces** **[Shen et al., Sun et al., Zhang et al.]**

2022

Exploit **hypergraphs** **[Wei et al., Xia et al.]**

2022

Use **graph attention networks** to recognize **meaningful** user-item **interactions at finer-grained** **[Zhang et al.]** level

Reproducibility and graph collaborative filtering

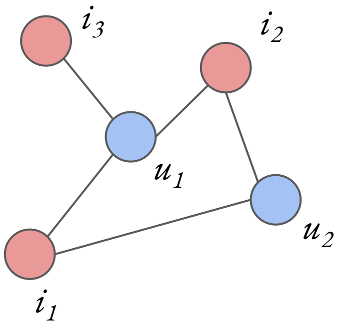
- **Reproducibility** in **machine learning** research is the **cutting-edge task** involving the **replication** of **experimental results** under the **same share settings** **[Bellogín and Said, Anelli et al. (2021a-2022), Ferrari Dacrema et al. (2019-2021), Sun et al.]**
- In **graph collaborative filtering**, **reproducibility** is **not** always **feasible** since **novel approaches** usually tend to
  - **copy and paste previous results** from the baselines
  - do **not provide full details** about the **experimental settings**
- What the **research community** should **seek to**
  - provide **more detailed descriptions** of the **experimental settings**
  - establish **standard evaluation metrics** and **experimental protocols**

Research questions

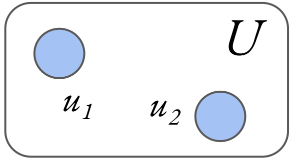
- **RQ1.** Is the **state-of-the-art** (i.e., the six most important papers) of **graph collaborative filtering** (graph CF) **replicable**?
- **RQ2.** How does the **state-of-the-art** of **graph CF position** with respect to **classic CF** state-of-the-art?
- **RQ3.** How does the **state-of-the-art** of **graph CF** perform **on datasets** from **different domains** and with **different topological** aspects, **not commonly adopted** for graph CF recommendation?
- **RQ4.** What **information (or lack of it)** impacts the **performance** of the **graph CF** methods across the **various datasets**?

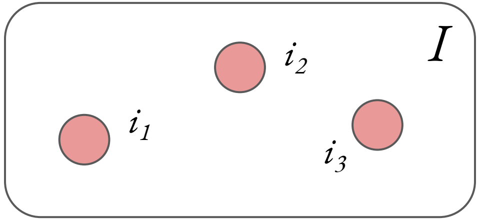
Background and reproducibility analysis

Background notions



Bipartite and undirected user-item graph





R

1	1	1
1	1	0

User-item interaction matrix

A

0	0	1	1	1
0	0	1	1	0
1	1	0	0	0
1	1	0	0	0
1	0	0	0	0

Adjacency matrix

Selected graph-based recommender systems

Model	Venue	Year	Strategy
NGCF	SIGIR	2019	• Pioneer approach in graph CF • Inter-dependencies among *ego* and *neighbor* nodes
DGCF	SIGIR	2020	• Disentangles users' and items' into intents and weights their importance • Updates graph structure according to those learned intents
LightGCN	SIGIR	2020	• Lightens the graph convolutional layer • Removes feature transformation and non-linearities
SGL	SIGIR	2021	• Brings self-supervised and contrastive learning to recommendation • Learns multiple node views through node/edge dropout and random walk
UltraGCN	CIKM	2021	• Approximates infinite propagation layers through a constraint loss and negative sampling • Explores item-item connections
GFCF	CIKM	2021	• Questions graph convolution in recommendation through graph signal processing • Proposes a strong close-form algorithm

Analysis on reported baselines

Families	Baselines	NGCF [71]	DGCF [73]	LightGCN [28]	SGL [78]	UltraGCN [47]	GFCF [59]
		Models					
		Used as graph CF baseline in (2021 — present)					
		[10, 13, 32, 62, 77, 84]	[19, 39, 46, 74, 75, 92]	[40, 54, 78, 82, 88, 89]	[22, 46, 77, 82, 85, 93]	[17, 24, 42, 48, 95, 96]	[4, 5, 41, 50, 80, 96]
Classic CF	MF-BPR [55]	✓	✓			✓	
	NeuMF [29]	✓					
	CMN [18]	✓					
	MacridVAE [44]		✓				
	Mult-VAE [38]			✓	✓		✓
	DNN+SSL [86]				✓		
	ENMF [11]					✓	
	CML [30]					✓	
	DeepWalk [52]					✓	
	LINE [66]					✓	
	Node2Vec [25]					✓	
	NBPO [91]					✓	

- Most of the **approaches** (apart from UltraGCN) are **compared against** a **small subset** of **classical CF** solutions
- The **recent literature** has raised **concerns** about **usually-untested strong CF** baselines **[Anelli et al. (2021a-2022), Ferrari Dacrema et al. (2019-2021), Zhu et al.]**

Analysis on reported baselines (cont.)

Families	**Baselines**	**Models**					
		NGCF [71]	DGCF [73]	LightGCN [28]	SGL [78]	UltraGCN [47]	GFCF [59]
		Used as graph CF baseline in (2021 — present)					
		[10, 13, 32, 62, 77, 84]	[19, 39, 46, 74, 75, 92]	[40, 54, 78, 82, 88, 89]	[22, 46, 77, 82, 85, 93]	[17, 24, 42, 48, 95, 96]	[4, 5, 41, 50, 80, 96]
	HOP-Rec [83]	✓					
	GC-MC [68]	✓	✓				
	PinSage [87]	✓					
	NGCF [71]		✓	✓	✓	✓	✓
	DisenGCN [43]		✓				
Graph CF	GRMF [53]			✓			✓
	GRMF-Norm [28]			✓			✓
	NIA-GCN [64]					✓	
	LightGCN [28]				✓	✓	✓
	DGCF [73]					✓	
	LR-GCCF [14]					✓	
	SCF [94]					✓	
	BGCF [63]					✓	
	LCFN [90]					✓	

- Conversely, most of the **approaches** are **compared against graph CF** solutions
- Orange ticks indicate that **no extensive comparison** among **the selected baselines** is performed (for **chronological** reasons)

Analysis on reported datasets

Models	Gowalla	Yelp 2018	Amazon Book	Alibaba-iFashion	Movielens 1M	Amazon Electronics	Amazon CDs
NGCF	✓	✓	✓				
DGCF	✓	✓	✓				
LightGCN	✓	✓	✓				
SGL		✓	✓	✓			
UltraGCN	✓	✓	✓		✓	✓	✓
GFCF	✓	✓	✓				

- Only a **limited subset** of **shared** recommendation **datasets**
- We include **novel**, never-investigated **datasets**

Analysis on experimental comparison

- **NGCF** train all baselines **from scratch**
- **DGCF** reports the results **directly from** the **NGCF** paper for the **shared baselines**
- **LightGCN**, **SGL**, and **UltraGCN copy and paste** from the original papers
- **GFCF reproduce** the results from **LightGCN** as the baselines are **exactly the same**
- Some **authors are shared** across such works

What we have done

- **Re-implement** from **scratch** all baselines by **carefully following** the **original works**
- **Train/evaluate** them within **Elliot** **[Anelli et al. (2021b), Malitesta et al. (2023a)]**
- Our goal is to **provide** a **fair** and **repeatable** experimental **environment**
- Use the **same hyper-parameter** settings as reported in the **original papers and codes**

Replication of prior results (RQ1)

Settings

- All **approaches** (**except for SGL**) use the **same datasets filtering** and **splitting** (**80/20 hold-out** splitting **user-wise**)
- **10% of the training** is left for **validation** for the **tuning of hyper-parameters** (**no indication** in the papers and/or codes)
- **All unrated items** as evaluation protocol
- Evaluation through the **Recall@20** and **nDCG@20** (**Recall@20** as **validation metric**)
- The **best settings** of **hyper-parameters** are usually **shared** in the **paper** and/or **code**

Results

Datasets	Models	Ours		Original		Performance Shift	
		Recall	nDCG	Recall	nDCG	Recall	nDCG
Gowalla	NGCF	0.1556	0.1320	0.1569	0.1327	$-1.3 \cdot 10^{-03}$	$-7 \cdot 10^{-04}$
	DGCF	0.1736	0.1477	0.1794	0.1521	$-5.8 \cdot 10^{-03}$	$-4.4 \cdot 10^{-03}$
	LightGCN	0.1826	0.1545	0.1830	0.1554	$-4 \cdot 10^{-04}$	$-9 \cdot 10^{-04}$
	SGL*	—	—	—	—	—	—
	UltraGCN	0.1863	0.1580	0.1862	0.1580	$+1 \cdot 10^{-04}$	0
	GFCF	0.1849	0.1518	0.1849	0.1518	0	0
Yelp 2018	NGCF	0.0556	0.0452	0.0579	0.0477	$-2.3 \cdot 10^{-03}$	$-2.5 \cdot 10^{-03}$
	DGCF	0.0621	0.0505	0.0640	0.0522	$-1.9 \cdot 10^{-03}$	$-1.7 \cdot 10^{-03}$
	LightGCN	0.0629	0.0516	0.0649	0.0530	$-2 \cdot 10^{-03}$	$-1.4 \cdot 10^{-03}$
	SGL	0.0669	0.0552	0.0675	0.0555	$-6 \cdot 10^{-04}$	$-3 \cdot 10^{-04}$
	UltraGCN	0.0672	0.0553	0.0683	0.0561	$-1.1 \cdot 10^{-03}$	$-8 \cdot 10^{-04}$
	GFCF	0.0697	0.0571	0.0697	0.0571	0	0
Amazon Book	NGCF	0.0319	0.0246	0.0337	0.0261	$-1.8 \cdot 10^{-03}$	$-1.5 \cdot 10^{-03}$
	DGCF	0.0384	0.0295	0.0399	0.0308	$-1.5 \cdot 10^{-03}$	$-1.3 \cdot 10^{-03}$
	LightGCN	0.0419	0.0323	0.0411	0.0315	$+8 \cdot 10^{-04}$	$+8 \cdot 10^{-04}$
	SGL	0.0474	0.0372	0.0478	0.0379	$-4 \cdot 10^{-04}$	$-7 \cdot 10^{-04}$
	UltraGCN	0.0688	0.0561	0.0681	0.0556	$+7 \cdot 10^{-04}$	$+5 \cdot 10^{-04}$
	GFCF	0.0710	0.0584	0.0710	0.0584	0	0

**Results are not provided since SGL was not originally trained and tested on Gowalla.*

- The **most significant** performance **shift** is in the order of **10^{-3}**
- **GFCF is the best replicated** one (no random initialization of model weights)
- **NGCF and DGCF rarely achieve 10^{-4}** because of the **random initializations** and **stochastic learning processes** involved
- **Replicability is ensured** and the **copy-paste** practise did **not hurt the results**

Benchmarking graph CF approaches using alternative baselines (RQ2)

Settings

- **Expand** the investigation to **four classic CF** recommender systems: **UserkNN**, **ItemkNN**, **RP³β**, **EASE^R** [Ferrari Dacrema et al. (2019), Anelli et al. (2022)]
- Consider two **unpersonalized** approaches (**MostPop** and **Random**)
- Follow the exact **same 80/20 train/test** splitting, and retain **our version** of the **10%** of the training as **validation**
- Use **Tree-structured Parzen Estimator** (with **20 exploration**) [Bergstra et al.]
- **Recall@20** is used as **validation metric**

Results

Families	Models	Gowalla		Yelp 2018		Amazon Book	
		Recall	nDCG	Recall	nDCG	Recall	nDCG
Reference	MostPop	0.0416	0.0316	0.0125	0.0101	0.0051	0.0044
	Random	0.0005	0.0003	0.0005	0.0004	0.0002	0.0002
Classic CF	UserkNN	0.1685	0.1370	0.0630	0.0528	0.0582	0.0477
	ItemkNN	0.1409	0.1165	0.0610	0.0507	0.0634	0.0524
	$\mathrm{RP}^3\beta$	0.1829	0.1520	0.0671	0.0559	0.0683	0.0565
	EASE^R*	0.1661	0.1384	0.0655	0.0552	**0.0710**	0.0567
Graph CF	NGCF	0.1556	0.1320	0.0556	0.0452	0.0319	0.0246
	DGCF	0.1736	0.1477	0.0621	0.0505	0.0384	0.0295
	LightGCN	0.1826	0.1545	0.0629	0.0516	0.0419	0.0323
	SGL	—	—	0.0669	0.0552	0.0474	0.0372
	UltraGCN	**0.1863**	**0.1580**	0.0672	0.0553	0.0688	0.0561
	GFCF	0.1849	0.1518	**0.0697**	**0.0571**	**0.0710**	**0.0584**

**Results for EASE^R on Amazon Book are taken from BARS Benchmark.*

- **Neither MostPop nor Random** get acceptable results: **popularity bias is not present in the datasets** or was removed (see later)
- Some of the **classic CF** approaches reach **better performance** than some **graph CF** baselines, and on **Yelp 2018** and **Amazon Book** they reach **best or second-to-best** performance

Extending the experimental comparison to new datasets (RQ3 - RQ4)

Settings

Statistics	Gowalla	Yelp 2018	Amazon Book	Allrecipes	BookCrossing
Users	29,858	31,668	52,643	10,084	6,754
Items	40,981	38,048	91,599	8,407	13,670
Edges	810,128	1,237,259	2,380,730	80,540	234,762
Density	0.0007	0.0010	0.0005	0.0010	0.0025
Avg. Deg. (U)	27.1327	39.0697	45.2241	7.9869	34.7590
Avg. Deg. (I)	19.7684	32.5184	25.9908	9.5801	17.1735

- Two **novel datasets**: **Allrecipes** and **BookCrossing** with **discordant characteristics** compared to the other datasets
- **Allrecipes**:
  - **users are more numerous** than items
  - **much lower average** user and item **degrees**
- **BookCrossing**:
  - **lowest ratio** between users and items
  - **much higher density** than the other datasets
- **Useful to assess** the performance in **different** (and **never-explored**) **topological** settings
- Use the **same experimental** setting from **RQ2** but with **validation set** (10% of the training set)

Results

Families	Models	Allrecipes		BookCrossing	
		Recall	nDCG	Recall	nDCG
Reference	MostPop	0.0472	0.0242	0.0352	0.0319
	Random	0.0024	0.0010	0.0013	0.0011
Classic CF	UserkNN	0.0339	0.0188	0.0871	0.0769
	ItemkNN	0.0326	0.0180	0.0779	0.0739
	$RP^3\beta$	0.0170	0.0089	**0.0941**	0.0821
	$EASE^R$	0.0351	0.0192	0.0925	**0.0847**
Graph CF	NGCF	0.0291	0.0144	0.0670	0.0546
	DGCF	0.0448	0.0234	0.0643	0.0543
	LightGCN	0.0459	0.0236	0.0803	0.0660
	SGL	0.0365	0.0192	0.0716	0.0600
	UltraGCN	**0.0475**	**0.0248**	0.0800	0.0651
	GFCF	0.0101	0.0051	0.0819	0.0712

- **Classic CF** methods are **very competitive**
- Especially on **BookCrossing**, the **classic CF** baselines are the **top-performing** approaches
- Only **UltraGCN and LightGCN** keep their **performance** as observed in the previous datasets
- For the **other graph-based** ones, the **performance** significantly **drops**

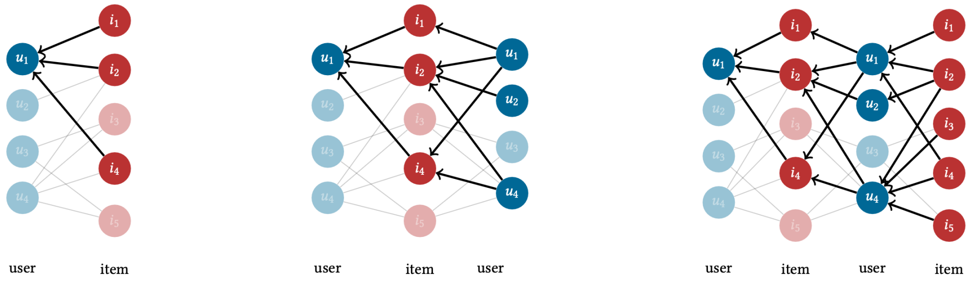
Discussion (graph-based models' ranking)

Metric		Gowalla	Yelp 2018	Amazon Book	Allrecipes	BookCrossing
Recall	**1.**	UltraGCN *(+19.73%)*	GFCF *(+25.36%)*	GFCF *(+122.57%)*	UltraGCN *(+370.30%)*	GFCF *(+27.37%)*
	2.	GFCF *(+18.83%)*	UltraGCN *(+20.86%)*	UltraGCN *(+115.67%)*	LightGCN *(+354.46%)*	LightGCN *(+24.88%)*
	3.	LightGCN *(+17.35%)*	SGL *(+20.32%)*	SGL *(+48.59%)*	DGCF *(+343.56%)*	UltraGCN *(+24.42%)*
	4.	DGCF *(+11.57%)*	LightGCN *(+13.13%)*	LightGCN *(+31.35%)*	SGL *(+261.39%)*	SGL *(+11.35%)*
	5.	NGCF (—)	DGCF *(+11.69%)*	DGCF *(+20.38%)*	NGCF *(+188.12%)*	NGCF *(+4.20%)*
	6.	*SGL** (—)	NGCF (—)	NGCF (—)	GFCF (—)	DGCF (—)
nDCG	**1.**	UltraGCN *(+19.70%)*	GFCF *(+26.33%)*	GFCF *(+137.40%)*	UltraGCN *(+386.27%)*	GFCF *(+31.12%)*
	2.	LightGCN *(+17.05%)*	UltraGCN *(+22.35%)*	UltraGCN *(+128.05%)*	LightGCN *(+362.75%)*	LightGCN *(+21.55%)*
	3.	GFCF *(+15.00%)*	SGL *(+22.12%)*	SGL *(+51.22%)*	DGCF *(+358.82%)*	UltraGCN *(+19.89%)*
	4.	DGCF *(+11.89%)*	LightGCN *(+14.16%)*	LightGCN *(+31.30%)*	SGL *(+276.47%)*	SGL *(+10.50%)*
	5.	NGCF (—)	DGCF *(+11.73%)*	DGCF *(+19.92%)*	NGCF *(+182.35%)*	NGCF *(+0.55%)*
	6.	*SGL** (—)	NGCF (—)	NGCF (—)	GFCF (—)	DGCF (—)

**SGL is not classifiable on the Gowalla dataset as results were not calculated in the original paper.*

- **UltraGCN and GFCF** are the two **best-performing** approaches
- All the **other approaches rank** according to the **chronological** order
- On **Allrecipes** and **BookCrossing**
  - **UltraGCN** preserves its role of **leading** approach
  - **GFCF** and **DGCF** performance is very **fluctuating**
  - **LightGCN** is in the **top positions** and **surpasses** other models which **should ideally outperform** it (e.g., SGL)
  - **NGCF poor performance** is confirmed

Discussion (analysis on the node degree)



(a) 1-hop (b) 2-hop (c) 3-hop

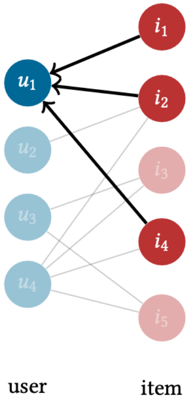
- We reinterpret node degree as **information flow from neighbor nodes to the ego nodes** after **multiple hops**
- Only **users as ending nodes** because a**ccuracy metrics** are calculated **user-wise**
- Information flow at **1, 2, and 3 hops**:

$$\boxed{\Upsilon^{(1)}_{\mathcal{U}}} = \mathbf{R}\mathbf{1}_{\mathcal{I}}, \qquad \Upsilon^{(2)}_{\mathcal{U}} = (\mathbf{R}\odot(\mathbf{1}_{\mathcal{U}}\mathbf{R}))\boxed{\mathbf{1}_{\mathcal{I}}}, \qquad \Upsilon^{(3)}_{\mathcal{U}} = (\mathbf{R}\mathbf{R}^{\top}\odot\mathbf{R}\mathbf{1}_{\mathcal{I}})\mathbf{1}_{\mathcal{I}},$$

column vector

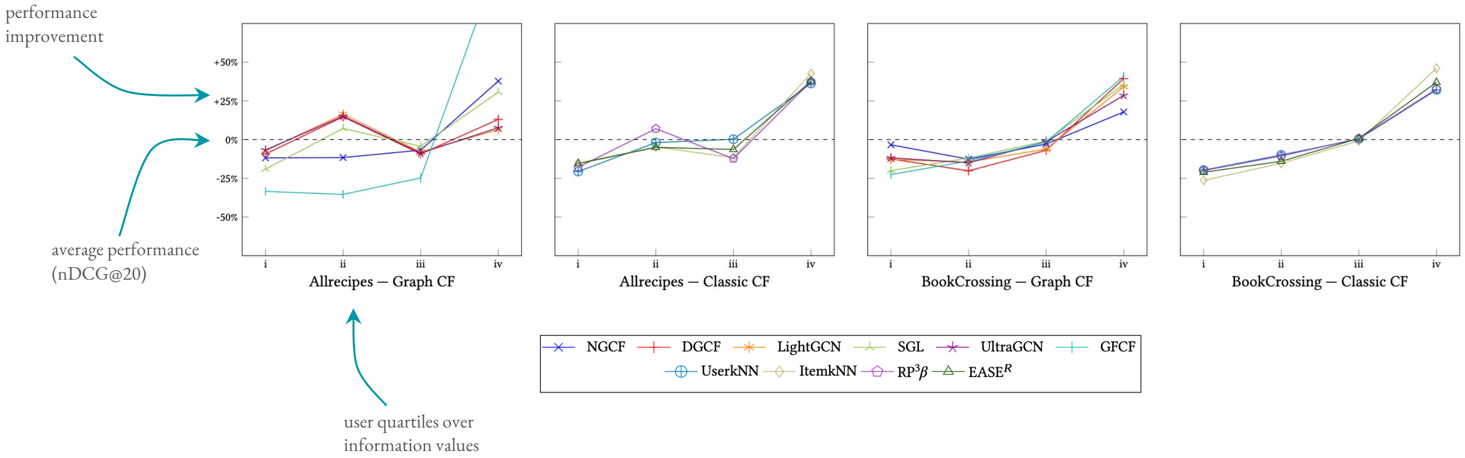
information after 1-hop

Analysis on the node degree (1-hop)



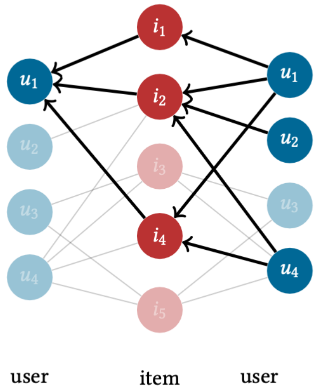
(a) 1-hop

Indication of the activeness of users on the platform

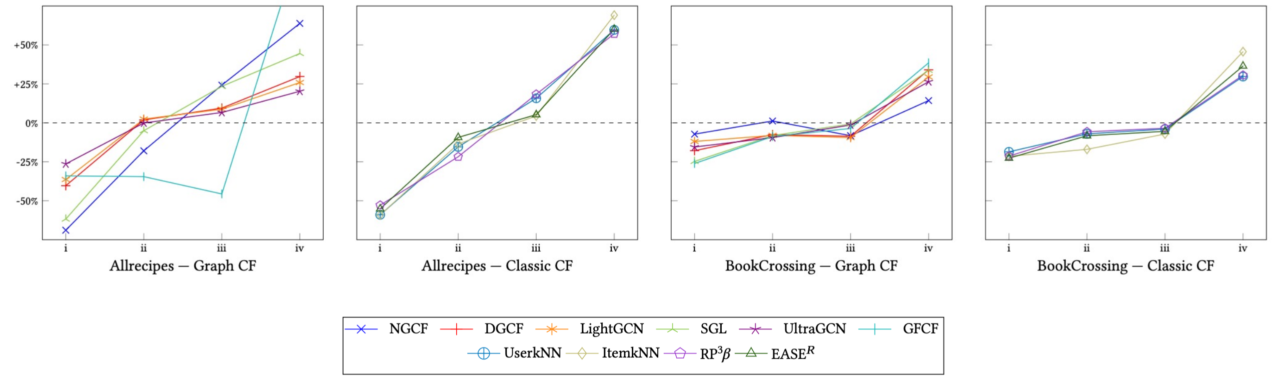


- The **4th quartile** is **favoured** with respect to the other ones
- The **trend** is even **more evident** on **GFCF**

Analysis on the node degree (2-hop)



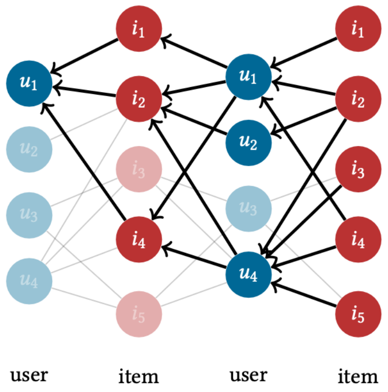
(b) 2-hop



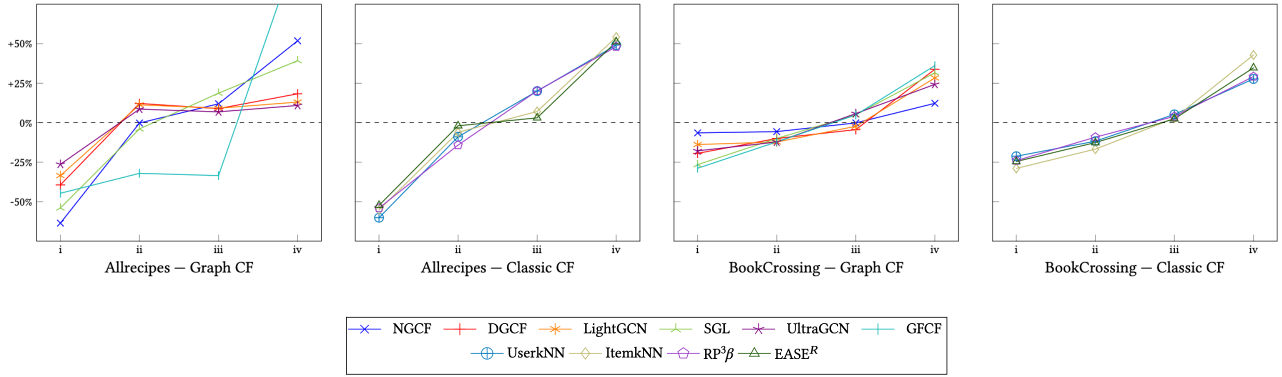
Indication of the influence of items' popularity on users

- Models **favour the warm users** who enjoyed **popular items** over the cold users who enjoyed niche items
- On **Allrecipes, UltraGCN, DGCF, and LightGCN** show **less discriminatory behavior** across quartiles; **SGL and NGCF** show a **higher slope** that is **comparable to classic CF** methods; **GFCF** behavior is even **more accentuated** than the 1-hop setting
- On **BookCrossing**, the trend is **almost aligned** across all models

Analysis on the node degree (3-hop)



(c) 3-hop



Indication of the influence of co-interacting users' activeness on users

- On **Allrecipes**, **UltraGCN, DGCF, and LightGCN** exhibit **more consistency** across quartiles, while **NGCF, SGL, and GFCF** have a more **disparate range** of results
- On **BookCrossing**, the information at the **3-hop is not providing** more insights than the 2-hop

Conclusion and future work

Conclusion

- **Replicate** the results of **six state-of-the-art graph CF** methods
- We include **other state-of-the-art approaches** and other (unexplored) **datasets**
- The **topological** graph **characteristics** (i.e., **node degree**) may impact the performance
- This happens **especially** for the **information flow at 2-hop** (i.e., user activeness + item popularity)

Future work

- **Further investigation** into **diversity** and **fairness** of graph CF approaches
- Analyze the **impact of other topological** graph characteristics on the performance (currently on arXiv **[Malitesta et al. (2023b)]**)

Useful resources

README.md

Graph-RSs-Reproducibility

This is the official repository for the paper "*Challenging the Myth of Graph Collaborative Filtering: a Reasoned and Reproducibility-driven Analysis*", accepted at RecSys 2023 (Reproducibility Track).

This repository is heavily dependent on the framework **Elliot**, so we suggest you refer to the official GitHub page and documentation.

Pre-requisites

We implemented and tested our models in `PyTorch==1.12.0`, with CUDA `10.2` and cuDNN `8.0`. Additionally, some of graph-based models require `PyTorch Geometric`, which is compatible with the versions of CUDA and `PyTorch` we indicated above.

Installation guidelines: scenario #1

If you have the possibility to install CUDA on your workstation (i.e., `10.2`), you may create the virtual environment with the requirement files we included in the repository, as follows:

```
# PYTORCH ENVIRONMENT (CUDA 10.2, cuDNN 8.0)

$ python3.8 -m venv venv
$ source venv/bin/activate
$ pip install --upgrade pip
$ pip install -r requirements.txt
$ pip install -r requirements_torch_geometric.txt
```

A Topology-aware Analysis of Graph Collaborative Filtering

A Topology-aware Analysis of Graph Collaborative Filtering

Daniele Malitesta
Politecnico di Bari
daniele.malitesta@poliba.it

Claudio Pomo
Politecnico di Bari
claudio.pomo@poliba.it

Vito W. Anelli
Politecnico di Bari
vitowalter.anelli@poliba.it

Alberto C. M. Mancino
Politecnico di Bari
alberto.mancino@poliba.it

Eugenio Di Sciascio
Politecnico di Bari
eugenio.disciascio@poliba.it

Tommaso Di Noia
Politecnico di Bari
tommaso.dinoia@poliba.it

[Malitesta et al. (2023b)]

References 1/2

[van den Berg et al.] 2017. *Graph convolutional matrix completion*. CoRR abs/1706.02263.

[Ying et al.] 2018. *Graph Convolutional Neural Networks for Web-Scale Recommender Systems*. In KDD. ACM, 974–983.

[Wang et al. (2019a)] 2019. *Neural Graph Collaborative Filtering*. In SIGIR. ACM, 165–174.

[Wang et al. (2019b)] 2019. *KGAT: Knowledge Graph Attention Network for Recommendation*. In KDD. ACM, 950–958.

[Chen et al.] 2020. *Revisiting Graph Based Collaborative Filtering: A Linear Residual Graph Convolutional Network Approach*. In AAAI. AAAI Press, 27–34.

[He et al.] 2020. *LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation*. In SIGIR. ACM, 639–648.

[Wang et al. (2020)] 2020. *Disentangled Graph Collaborative Filtering*. In SIGIR. ACM, 1001–1010.

[Tao et al.] 2020. *MGAT: Multimodal Graph Attention Network for Recommendation*. Inf. Process. Manag. 57, 5 (2020), 102277.

[Wu et al.] 2021. *Self-supervised Graph Learning for Recommendation*. In SIGIR. ACM, 726–735.

[Yu et al.] 2022. *Are Graph Augmentations Necessary?: Simple Graph Contrastive Learning for Recommendation*. In SIGIR. ACM, 1294–1303.

[Mao et al.] 2021. *UltraGCN: Ultra Simplification of Graph Convolutional Networks for Recommendation*. In CIKM. ACM, 1253–1262.

[Peng et al.] 2022. *SVD-GCN: A Simplified Graph Convolution Paradigm for Recommendation*. In CIKM. ACM, 1625–1634.

[Shen et al.] 2021. *How Powerful is Graph Convolution for Recommendation?*. In CIKM. ACM, 1619–1629.

[Sun et al.] 2021. *HGCF: Hyperbolic Graph Convolution Networks for Collaborative Filtering*. In WWW. ACM / IW3C2, 593–601.

[Zhang et al.] 2022. *Geometric Disentangled Collaborative Filtering*. In SIGIR. ACM, 80–90.

[Wei et al.] 2022. *Dynamic Hypergraph Learning for Collaborative Filtering*. In CIKM. ACM, 2108–2117.

[Xia et al.] 2022. *Hypergraph Contrastive Collaborative Filtering*. In SIGIR. ACM, 70–79.

[Bellogín and Said] 2021. *Improving accountability in recommender systems research through reproducibility*. User Model. User Adapt. Interact. 31, 5 (2021), 941–977.

[Anelli et al. (2021a)] 2021. *Reenvisioning the comparison between Neural Collaborative Filtering and Matrix Factorization*. In RecSys. ACM, 521–529.

[Anelli et al. (2022)] 2022. *Top-N Recommendation Algorithms: A Quest for the State-of-the-Art*. In UMAP. ACM, 121–131.

References 2/2

[Ferrari Dacrema et al. (2019)] 2019. *Are we really making much progress? A worrying analysis of recent neural recommendation approaches*. In RecSys. ACM, 101–109.

[Ferrari Dacrema et al. (2021)] 2021. *A Troubling Analysis of Reproducibility and Progress in Recommender Systems Research*. ACM Trans. Inf. Syst. 39, 2 (2021), 20:1–20:49.

[Sun et al.] 2020. *Are We Evaluating Rigorously? Benchmarking Recommendation for Reproducible Evaluation and Fair Comparison*. In RecSys. ACM, 23–32.

[Zhu et al.] 2022. *BARS: Towards Open Benchmarking for Recommender Systems*. In SIGIR. ACM, 2912–2923.

[Anelli et al. (2021b)] 2021. *Elliot: A Comprehensive and Rigorous Framework for Reproducible Recommender Systems Evaluation*. In SIGIR. ACM, 2405–2414.

[Malitesta et al. (2023a)] 2023. *An Out-of-the-Box Application for Reproducible Graph Collaborative Filtering extending the Elliot Framework*. In UMAP (Adjunct Publication). ACM, 12–15.

[Bergstra et al.] 2011. *Algorithms for Hyper-Parameter Optimization*. In NIPS. 2546–2554.

[Malitesta et al. (2023b)] 2023. *A Topology-aware Analysis of Graph Collaborative Filtering*. CoRR abs/2308.10778.