Soongsil University

LightGCL : SIMPLE YET EFFECTIVE GRAPH CONTRASTIVE LEARNING FOR RECOMMENDATION

CheolHee Jung

DM Lab

05.23.2024

Xuheng Cai Chao Huang Lianghao Xia Xubin Ren

ICLR'23

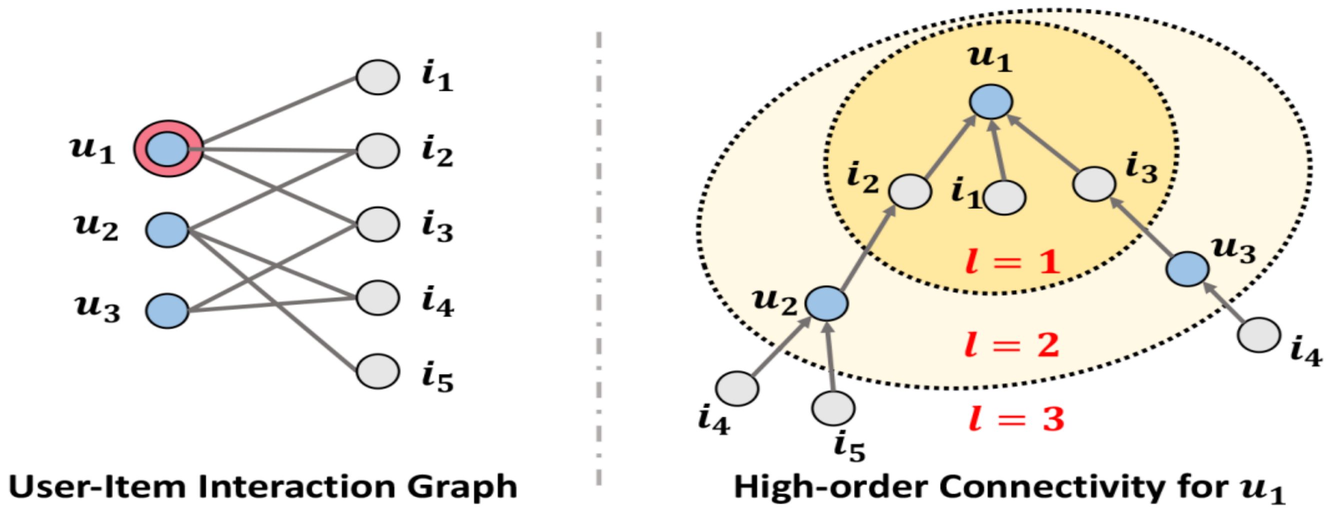
Outline

- Background
- Proposed Method
- Experiments
- Conclusion

LightGCN

- ❒ LightGCN 추천 시스템에서 GCN을 사용하는 가장 대표적인 방법, NGCF에서 기본적인 구조를 차용

- ❒ NGCF는 GCN을 활용하여 Collaborative Filtering을 수행한 방법론



- User – Item matrix를 이분할(User - Item) 그래프 형식으로 활용한 최초의 방법
- 개별 User / Item은 ID index를 통한 임베딩 값 가짐

Neural Graph Collaborative Filtering

LightGCN

❒ 일반적인 GCN식

$$e_u^{k+1} = \sigma(W_1 e_u^k + \sum_{i \in N_u} \frac{1}{\sqrt{|N_u||N_i|}} (W_1 e_i^k + W_2(e_i^k \odot e_u^k)))$$

$$e_i^{k+1} = \sigma(W_1 e_i^k + \sum_{i \in N_i} \frac{1}{\sqrt{|N_u||N_i|}} (W_1 e_u^k + W_2(e_u^k \odot e_i^k)))$$

- LightGCN 논문에서는 위 구조에서(non-linear activation, feature transform)의 효과에 대해 검증함
  - 각 요소의 값이 없을 때가 더 나은 성능을 보여줌

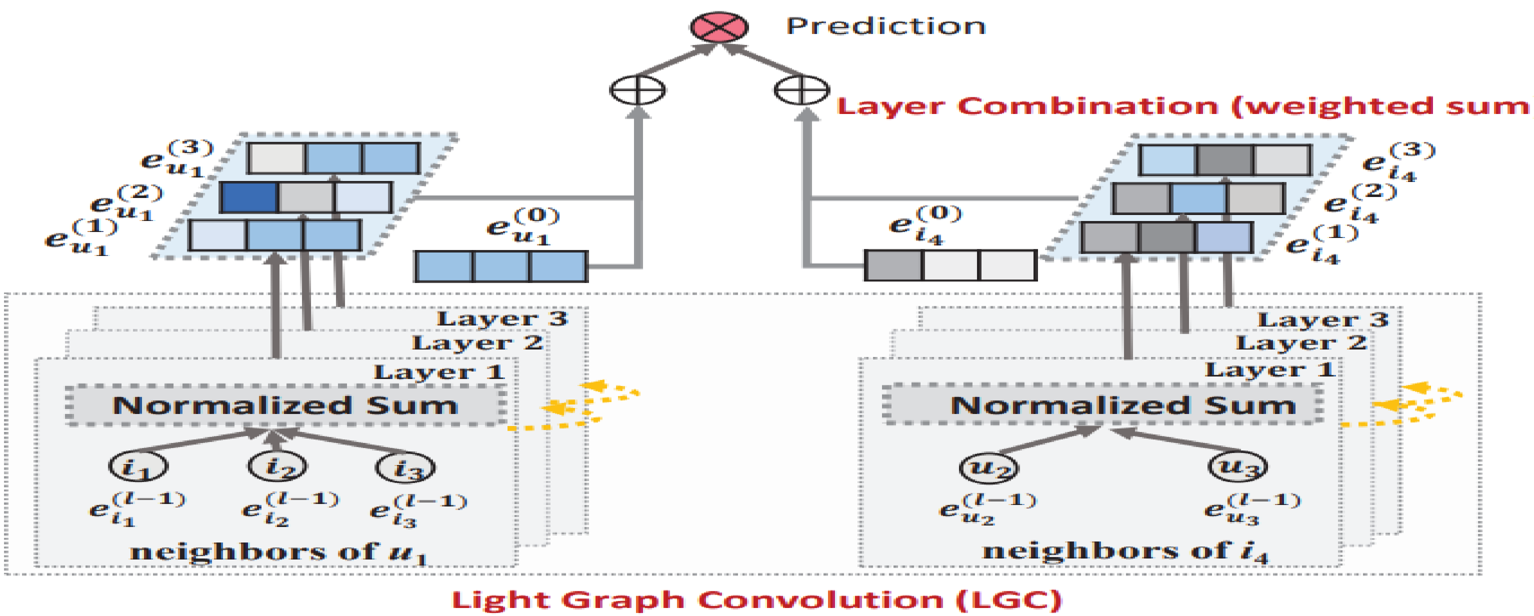
LightGCN

❒ 더 간략한 GCN 구조를 설계

$$e_u^{k+1} = \sum_{i \in N_u} \frac{1}{\sqrt{|N_u||N_i|}} e_i^k$$
$$e_i^{k+1} = \sum_{i \in N_i} \frac{1}{\sqrt{|N_i||N_u|}} e_u^k$$

- Feature transform에 따른 weight는 존재하지 않음
- Self – connection을 사용하지 않음
  - 이를 보완하기 위해 각 layer의 output값을 함께 사용

Self – Connection



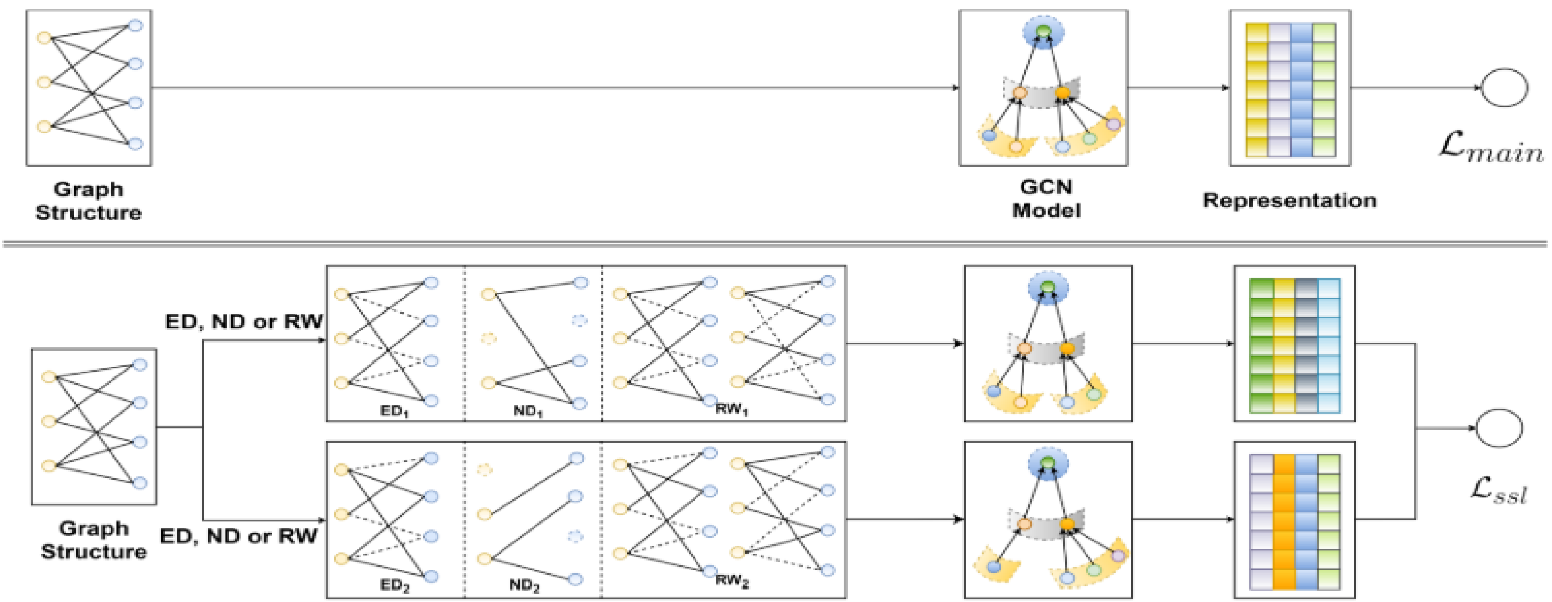
$$e_u = \sum_{k=0}^{K} \alpha_k e_u^k,\ \ e_i = \sum_{k=0}^{K} \alpha_k e_i^k,\ \ \hat{y}_{ui} = e_u^T e_i,$$

$$L_{BPR} = -\sum_{u=1}^{M} \sum_{i \in N_u} \sum_{j \notin N_u} \ln \sigma\left(\hat{y}_{ui} - \hat{y}_{uj}\right) + \lambda \|E^0\|^2$$

- 위 식 과정이 self – connection을 사용하는 것과 같은 의미
- 최종적으로 생성되는 item과 user 임베딩의 내적값을 통해 Score 도출

Graph Contrastive Learning

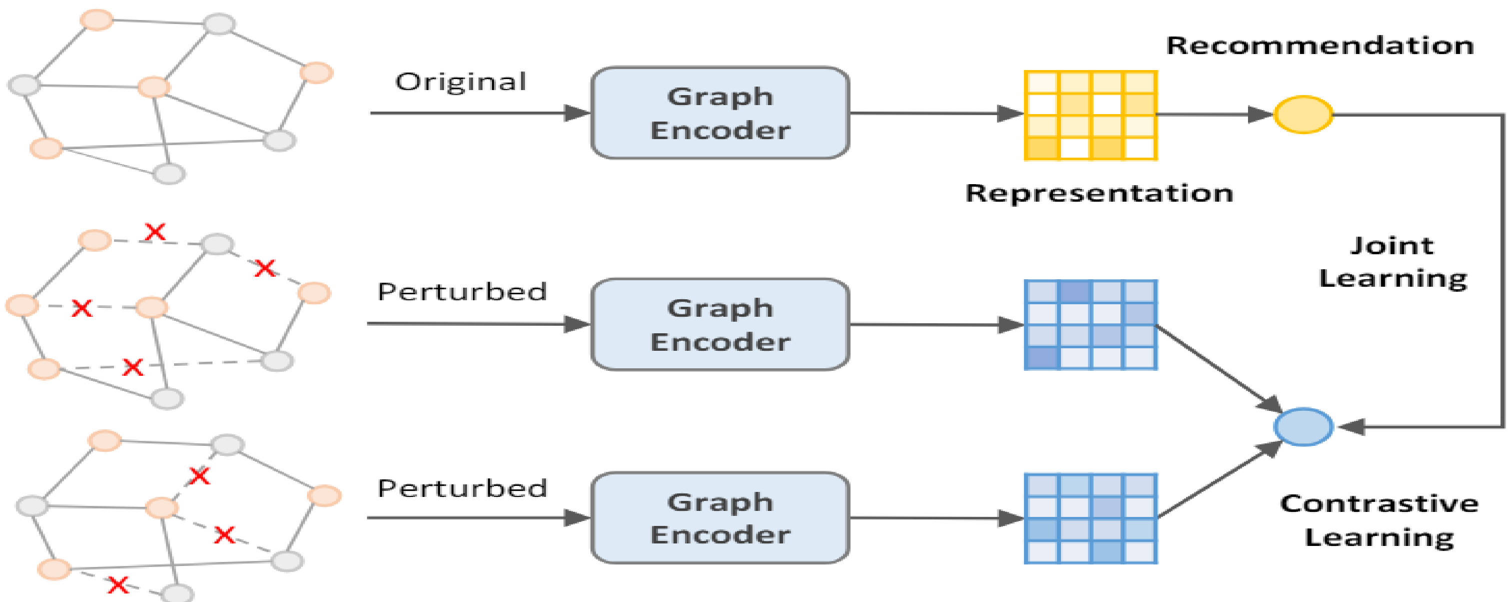
❐ Graph Contrastive Learning for Recommendation



- Representation 보다 Contrastive Loss를 제약조건
- Node drop, Edge drop, R.W 을 통한 Graph Augmentation
  - Sparsity 문제로 인한 편향된 학습 가능성 제시

Graph Contrastive Learning

❒ Are Graph Augmentations Necessary?



- 전통적인 Augmentation을 하지 않고 그래프 embedding에 perturbation추가

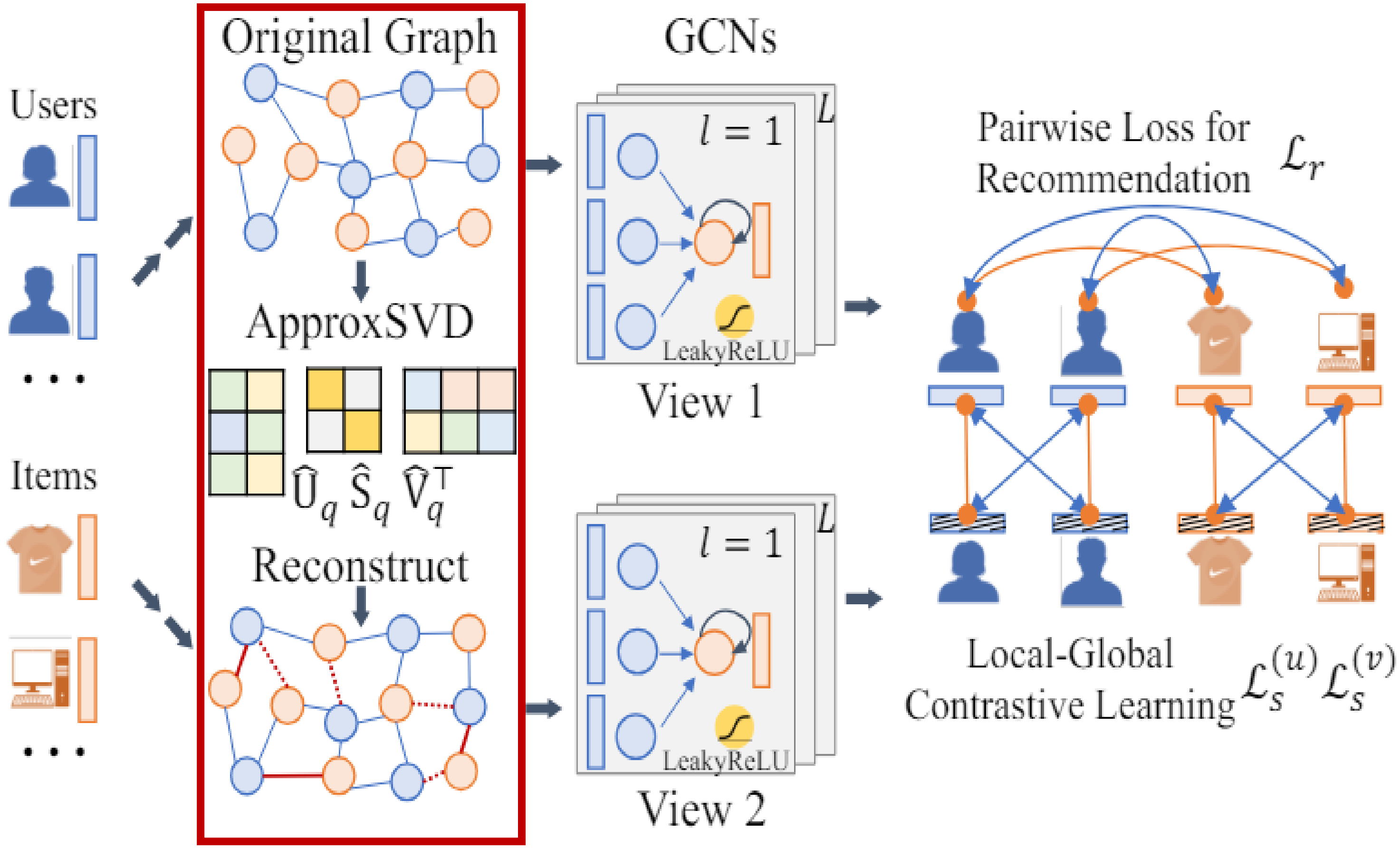
❒ Augmentation을 더 잘하면 성능이 좋지 않을까?

Outline

- Background
- Proposed Method
- Experiments
- Conclusion

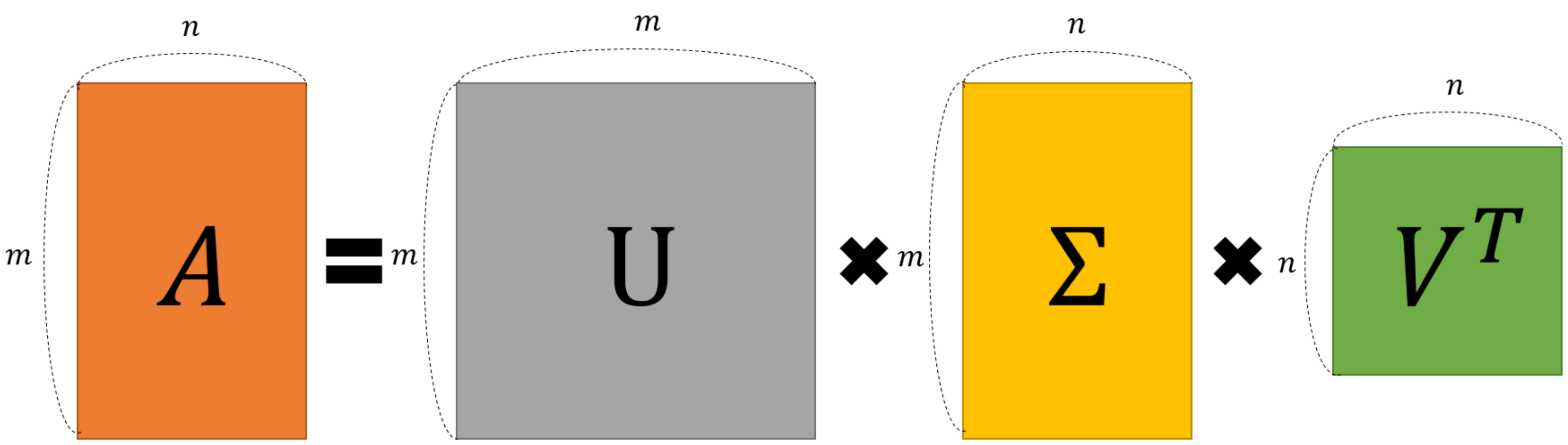
LightGCL

❒ Overall structure of LightGCL



LightGCL

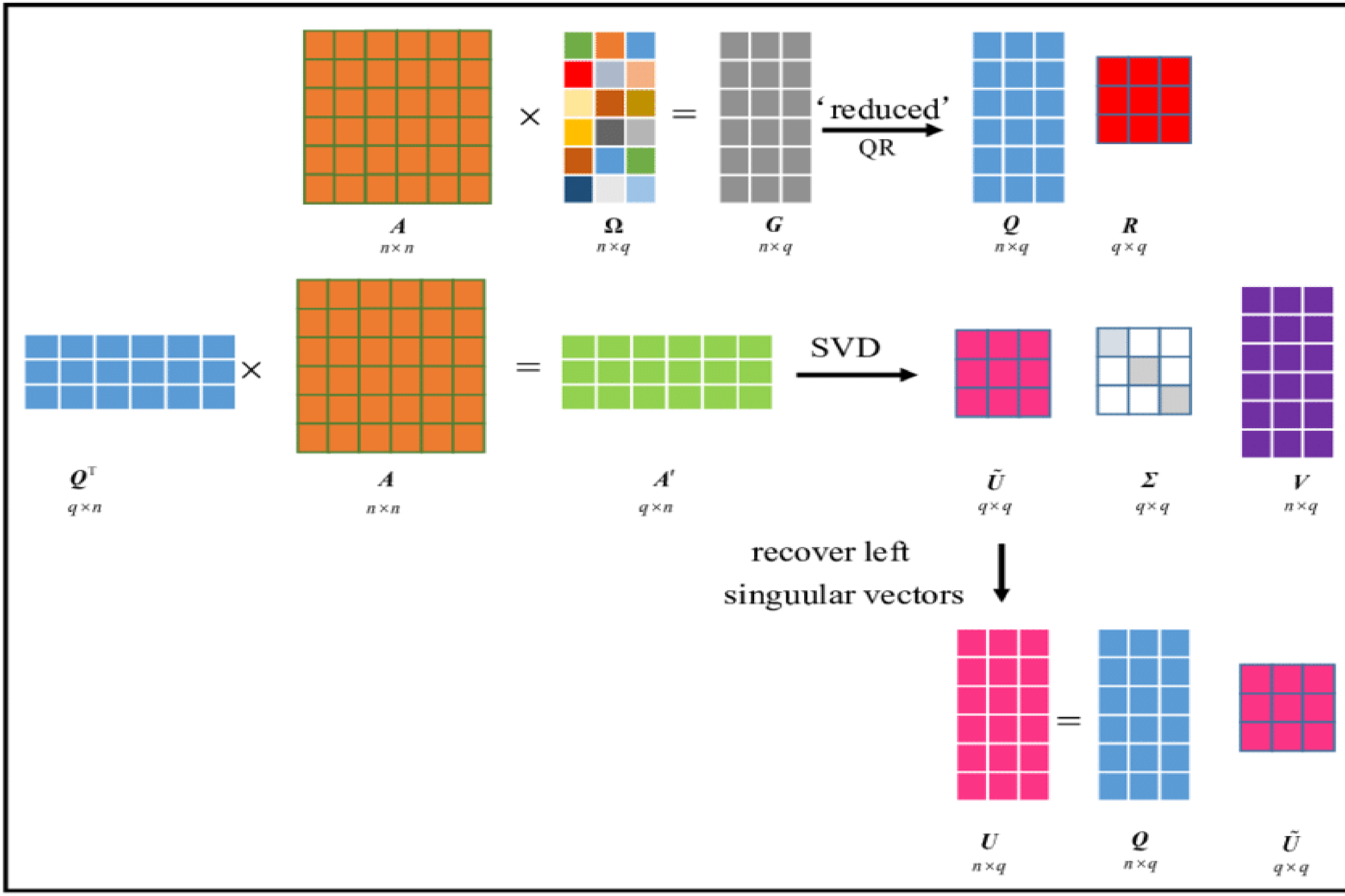
❒ Efficient Global Collaborative Relation Learning



- User와 Item의 정보를 가진 채, User– Item Pair를 복원
- 본 논문에서는 SVD는 Truncate 하여 차원을 축소해서 사용
- Large Adjaceny matrix에 바로 적용하기 때문에 계산량 많음 → Randomized SVD 사용

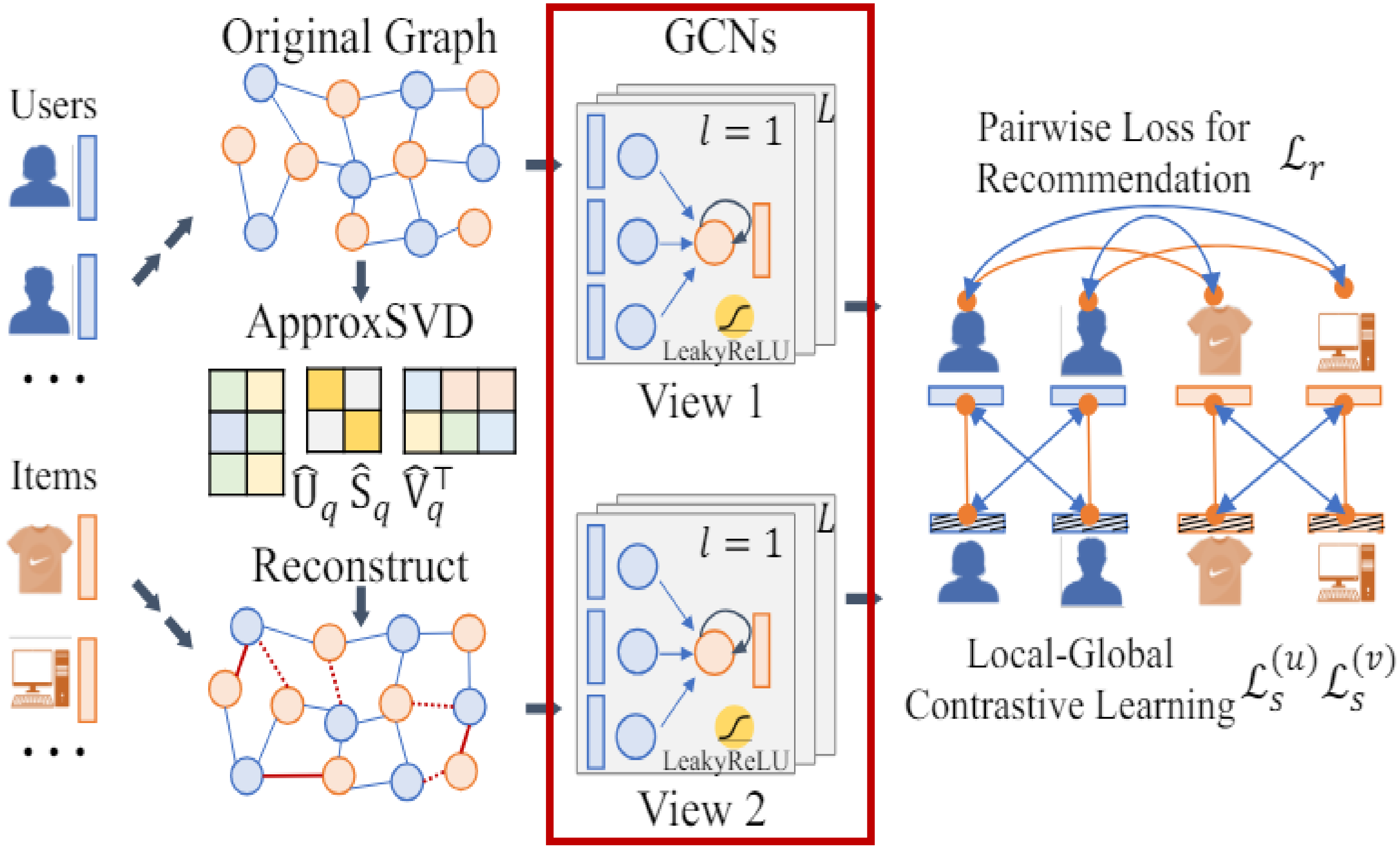
LightGCL

❒ Randomized SVD



LightGCL

❒ Overall structure of LightGCL



LightGCL

❒ Propagation & Aggregation

$$Z_{i,l}^{(u)} = \sigma(p(\tilde{A}_{i,:}) \cdot E_{l-1}^{(u)}, \quad Z_{j,l}^{(v)} = \sigma(p(\tilde{A}_{:,j}) \cdot E_{l-1}^{(u)}$$

- 개별 Item, User 임베딩 값에 Normalized Adjacency Mat 값을 통해 Aggregation됨

	Item1	Item2	Item3	Item4
User1	3	**5**	**5**	1
User2	**4**	**5**	**5**	2
User3	**5**	**4**	2	**4**
User4	3	**5**	**4**	3
User5	1	**4**	**5**	2

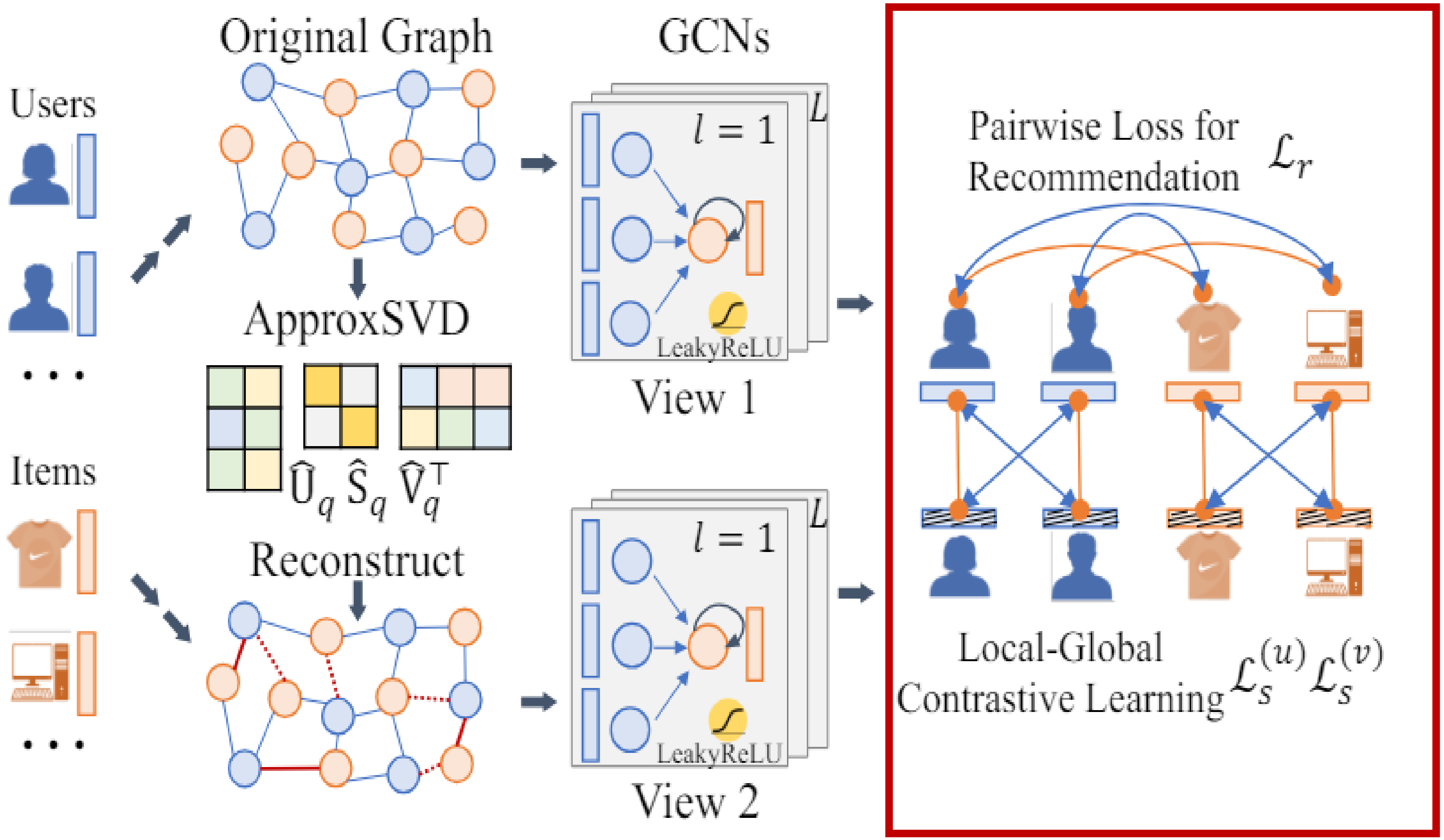
- Bi-partite graph 구조를 통해, User는 propagation 시 이웃 Item의 정보만 받음

❒ 최종 임베딩 계산

$$e_i^{(u)} = \sum_{l=0}^{L} Z_{i,l}^{(u)}, \quad e_j^{(v)} = \sum_{l=0}^{L} Z_{j,l}^{(v)}, \quad \hat{y}_{i,j} = e_i^{(u)T} e_j^{(v)}$$

LightGCL

❒ Overall structure of LightGCL



LightGCL

❒ Simplified Local-Global Contrastive Learning

- Contrastive loss

$$L_s^{(u)} = \sum_{i=0}^{I} \sum_{l=0}^{L} -\log \frac{\exp(s\left(z_{i,l}^{(u)}, g_{i,l}^{(u)}\right)/\tau}{\sum_{i'=0}^{I} \exp(s\left(z_{i,l}^{(u)}, g_{i',l}^{(u)}\right)/\tau}$$

  - User와 Item에 대해 각각 해당 Loss값 산출
  - Loss를 구하는 과정에서 Node drop, batch마다 진행

- Pairwise Loss

$$L_r = \sum_{i=0}^{I} \sum_{s=1}^{S} \max(0, 1 - \hat{y}_{i,p_s} + \hat{y}_{i,n_s})$$

  - Positive pair score가 Negative보다 크도록 유도

- Loss

$$L = L_r + \lambda_1 \cdot (L_s^{(u)} + L_s^{(v)}) + \lambda_2 \, \|\Theta\|_2^2$$

Outline

- Background
- Proposed Method
- Experiments
- Conclusion

Experiments

❐ Dataset

Dataset	User	Items	Interactions
Yelp	29,601	24,734	1,069,128
Gowalla	50,821	57,440	1,172,425
ML – 10M	69,878	10,195	6,999,171
Amazon – Book	78,578	77,801	2,240,156
Tmall	47,939	41,390	2,357,450

Experiments

Table 1: Performance comparison with baselines on five datasets.

Data	Metric	DGCF	HyRec	LightGCN	MHCN	SGL	SimGRACE	GCA	HCCF	SHT	SimGCL	LightGCL	*p-val.*	*impr.*
Yelp	R@20	0.0466	0.0472	0.0482	0.0503	0.0526	0.0603	0.0621	0.0626	0.0651	0.0718	**0.0793**	7e-9	10%
	N@20	0.0395	0.0395	0.0409	0.0424	0.0444	0.0435	0.0530	0.0527	0.0546	0.0615	**0.0668**	8e-9	8%
	R@40	0.0774	0.0791	0.0803	0.0826	0.0869	0.0989	0.1021	0.1040	0.1091	0.1166	**0.1292**	2e-9	10%
	N@40	0.0511	0.0522	0.0527	0.0544	0.0571	0.0656	0.0677	0.0681	0.0709	0.0778	**0.0852**	2e-9	9%
Gowalla	R@20	0.0944	0.0901	0.0985	0.0955	0.1030	0.0869	0.0896	0.1070	0.1232	0.1357	**0.1578**	1e-6	16%
	N@20	0.0522	0.0498	0.0593	0.0574	0.0623	0.0528	0.0537	0.0644	0.0731	0.0818	**0.0935**	2e-6	14%
	R@40	0.1401	0.1356	0.1431	0.1393	0.1500	0.1276	0.1322	0.1535	0.1804	0.1956	**0.2245**	3e-6	14%
	N@40	0.0671	0.0660	0.0710	0.0689	0.0746	0.0637	0.0651	0.0767	0.0881	0.0975	**0.1108**	3e-6	13%
ML-10M	R@20	0.1763	0.1801	0.1789	0.1497	0.1833	0.2254	0.2145	0.2219	0.2173	0.2265	**0.2613**	1e-9	15%
	N@20	0.2101	0.2178	0.2128	0.1814	0.2205	0.2686	0.2613	0.2629	0.2573	0.2613	**0.3106**	3e-9	18%
	R@40	0.2681	0.2685	0.2650	0.2250	0.2768	0.3295	0.3231	0.3265	0.3211	0.3345	**0.3799**	7e-10	13%
	N@40	0.2340	0.2340	0.2322	0.1962	0.2426	0.2939	0.2871	0.2880	0.3318	0.2880	**0.3387**	1e-9	17%
Amazon	R@20	0.0211	0.0302	0.0319	0.0296	0.0327	0.0381	0.0309	0.0322	0.0441	0.0474	**0.0585**	2e-7	23%
	N@20	0.0154	0.0225	0.0236	0.0219	0.0249	0.0291	0.0238	0.0247	0.0328	0.0360	**0.0436**	2e-6	21%
	R@40	0.0351	0.0432	0.0499	0.0489	0.0531	0.0621	0.0498	0.0525	0.0719	0.0750	**0.0933**	1e-7	24%
	N@40	0.0201	0.0246	0.0290	0.0284	0.0312	0.0371	0.0301	0.0314	0.0420	0.0451	**0.0551**	9e-7	22%
Tmall	R@20	0.0235	0.0233	0.0225	0.0203	0.0268	0.0222	0.0373	0.0314	0.0387	0.0473	**0.0528**	3e-5	11%
	N@20	0.0163	0.0160	0.0154	0.0139	0.0183	0.0152	0.0252	0.0213	0.0262	0.0328	**0.0361**	1e-4	10%
	R@40	0.0394	0.0350	0.0378	0.0340	0.0446	0.0367	0.0616	0.0519	0.0645	0.0766	**0.0852**	1e-5	11%
	N@40	0.0218	0.0199	0.0208	0.0188	0.0246	0.0203	0.0337	0.0284	0.0352	0.0429	**0.0473**	7e-5	10%

Experiments

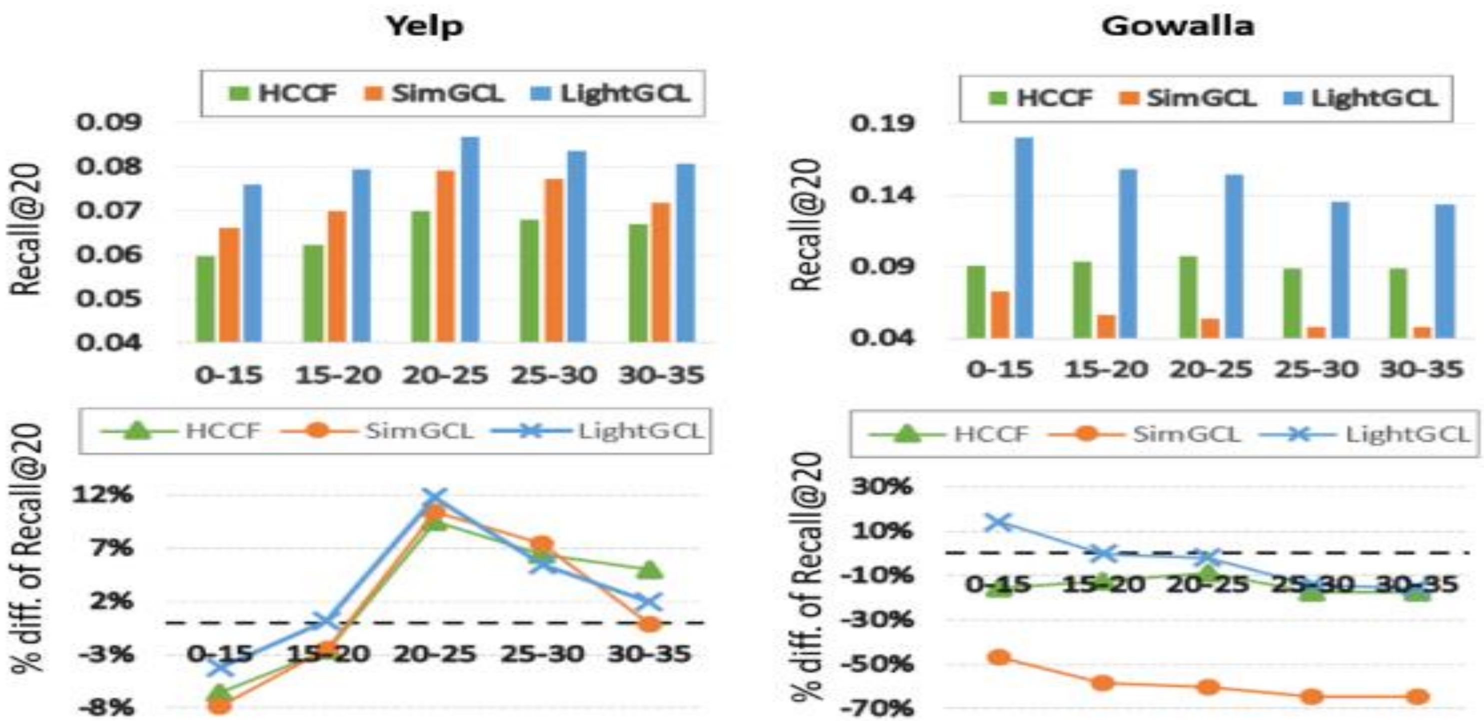


Figure 2: Performance on users of different sparsity degrees, in terms of *Recall* (histograms) and relative *Recall w.r.t* overall performances (charts).

- Interaction 이 적은 그룹에서 다른 방법 대비 성능이 상대적으로 우수

Experiments

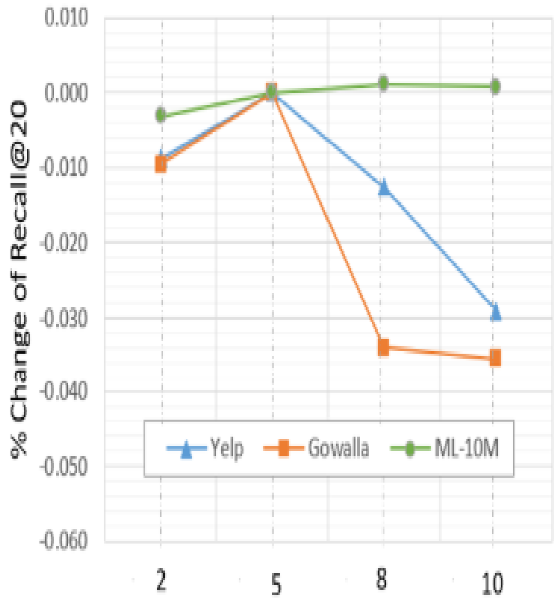


Table 4: Ablation study on LightGCL.

Variant	Yelp		Gowalla	
	Recall@20	NDCG@20	Recall@20	NDCG@20
CL-MF	0.0781	0.0659	0.1561	0.0929
CL-SVD++	0.0788	0.0666	0.1568	0.0932
LightGCL	**0.0793**	**0.0668**	**0.1578**	**0.0935**

- 다른 matrix decomposition 방법론을 적용하여 실험 진행
- Rank(SVD)에 따른 성능 변화

Experiments

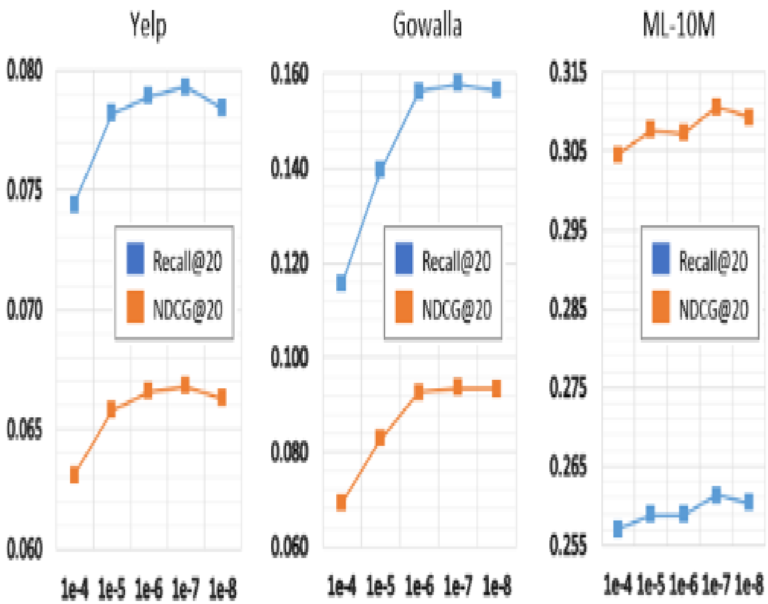


Figure 6: Impact of λ_1.

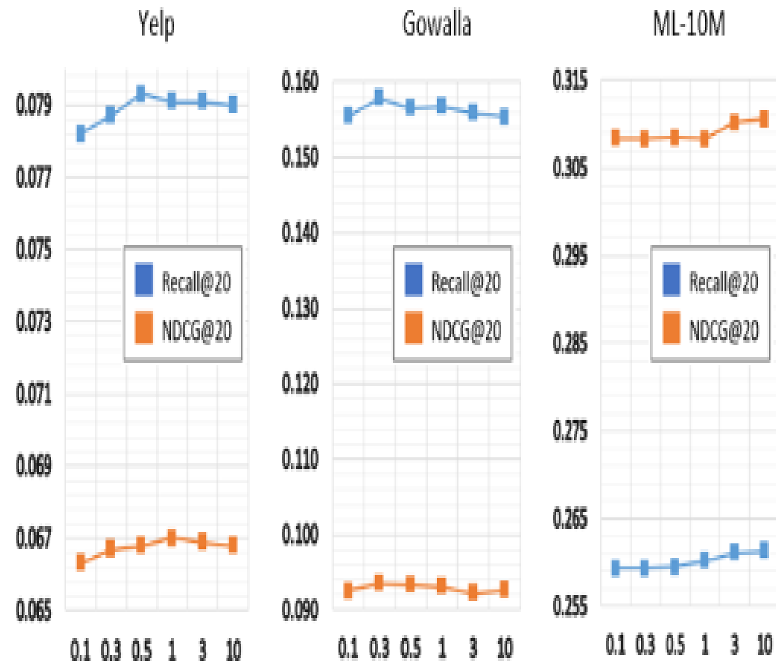


Figure 7: Impact of τ

$$L = L_r + \lambda_1 \cdot (L_s^{(u)} + L_s^{(v)}) + \lambda_2 \ ||\Theta||_2^2$$

Outline

- Background
- Proposed Method
- Experiments
- Conclusion

Conclusion

- LightGCL은 SVD 기반의 그래프 증강 기법을 제안하여, 기존의 한계를 극복하고 우수한 성능을 달성

- Weak points
  - Contrastive Learning 자체의 문제점
    - Link prediction을 잘 수행하지 못함

Thank you!

Q & A