

Ada-Comp: A Post-Retrain Strategy with Adaptive Compensation According to User Segments in Unlearning for GNN Recommender Systems

Eugene Jeong

eugene_jeong@dgist.ac.kr

Daegu Gyeongbuk Institute of Science and Technology
Daegu, Republic of Korea

Sang-chul Lee*

sangchul.lee@dgist.ac.kr

Daegu Gyeongbuk Institute of Science and Technology
Daegu, Republic of Korea

Abstract

User unlearning in GNN-based recommender systems can cause severe performance collapse and amplify group-wise disparity, especially when user deletion targets high-centrality G_{Niche} users. We refer to this phenomenon as a *bridge-collapse* pattern, in which deleting bridge-like users can disrupt graph structure and increase group-wise disparity. To mitigate the resulting group-wise degradation, we propose **Ada-Comp**, a diagnosis-driven post-retraining strategy with adaptive compensation that identifies the lowest-performing user group after unlearning and applies targeted fine-tuning. For NGCF after G_{Niche} deletion, Ada-Comp improves catalog coverage by 8.9% and reduces group-wise disparity by 10.3% relative to Retrain, while keeping overall NDCG within 4.6% of Retrain.

CCS Concepts

• **Information systems** → **Recommender systems**; • **Computing methodologies** → **Machine learning**; • **Security and privacy** → *Privacy-preserving protocols*.

Keywords

Machine Unlearning, Graph Neural Networks, Recommender Systems, Disparity, Structural Fragility, Adaptive Compensation

1 Introduction

User unlearning [1] removes the influence of users requesting deletion, motivated by the Right to be Forgotten [3]. GNN-based recommendation models represent user-item interactions as a bipartite graph and propagate signals over observed edges [4, 8, 10, 13]. In user unlearning, deletion requests remove the target users' interaction records, which can perturb the graph-dependent training signal before the model is updated through retraining or approximate unlearning procedures [1, 2, 11]. Recent graph-unlearning studies mainly focus on certified removal or verification of unlearning effectiveness in graph models [2, 11, 12], whereas our focus is on group-wise disparity and robustness after user deletion in GNN-based recommendation. We present a structural and group-aware analysis of GNN recommender systems for this setting. We use four behavior-based groups to track accuracy, group-wise disparity, and catalog coverage before and after unlearning. This reveals imbalance caused by graph perturbations [5, 6].

We hypothesize that initial GNN group-wise disparity arises from a structural bottleneck. We use the term structural bottleneck to

denote a structural constraint arising from dependence on bridge-like users. The *noisy-bridge hypothesis* posits that a high-centrality bridge-like group mediates propagation paths for structurally vulnerable users. At the same time, the signals propagated through this group can interfere with those users. Deleting such users breaks propagation paths and can trigger a *bridge-collapse* pattern, resulting in severe utility loss and amplified group-wise disparity after unlearning. We define the *bridge-collapse dilemma* as the tension between reducing interfering propagation from a bridge-like group and preserving the graph structure required for stable GNN unlearning. Motivated by these observations, we propose **Ada-Comp**, a post-retrain strategy with adaptive compensation. It first retrains the model after user deletion, then identifies the lowest-performing group from the post-unlearning performance profile and applies targeted fine-tuning to that group.

Our contributions are as follows:

- We formulate user unlearning in GNN-based recommendation as a group-sensitive structural problem, rather than only an accuracy recovery task.
- We identify the noisy-bridge effect, where signals propagated through bridge-like users can interfere with structurally vulnerable users.
- We develop **Ada-Comp**, a lightweight compensation strategy that adjusts the unlearned model according to the affected group.

2 Problem Definition

2.1 Definition of User Segmentation

We begin with the bridge-collapse dilemma, which arises from the tension between reducing interfering propagation from bridge-like users and preserving the graph structure required for stable GNN unlearning. We formulate the noisy-bridge hypothesis that signals propagated through bridge-like users can interfere with structurally vulnerable users. To examine this hypothesis, we partition users into four groups by their behavior: Sparse, Niche, Specialist, and Majority. Overlapping assignments are resolved so that each user belongs to exactly one of the four groups.

The Sparse group, G_{Sparse} , consists of users with very low activity density. We define the activity density of a user u as

$$\text{activity_density}(u) = \frac{N_u}{D_u}, \quad (1)$$

where N_u denotes the total number of feedback records given by user u , and D_u is the number of days between the timestamps of the first and last feedback records. Users whose $\text{activity_density}(u)$ falls within the bottom 20% of all users are classified into G_{Sparse} .

*Corresponding author.

The Niche group, G_{Niche} , consists of users with low average popularity of items that they consume. To identify this group, we define the average popularity of a user u , $\text{pop_mean}(u)$, as

$$\text{pop_mean}(u) = \frac{1}{N_u} \sum_{i \in I_u} \log(1 + \text{pop}(i)). \quad (2)$$

Here, I_u denotes the set of items rated by user u , and $N_u = |I_u|$ denotes its cardinality. Also, $\text{pop}(i)$ denotes the popularity of item i , computed as the number of user feedback records. The term $\log(1 + \text{pop}(i))$ is a logarithmic transformation used to mitigate the long-tail effect in the popularity distribution. Users whose $\text{pop_mean}(u)$ falls within the bottom 20% among all users are classified into G_{Niche} .

The Specialist group, $G_{\text{Specialist}}$, consists of active users whose interactions are highly concentrated on a small number of categories (genres). We employ the category entropy and the total interaction count. The category entropy is defined as

$$\text{category_entropy}(u) = - \sum_{c \in C_u} p_u(c) \log p_u(c), \quad (3)$$

where C_u is the set of genres consumed by user u , and $p_u(c)$ denotes the proportion of interactions belonging to genre c among all interactions of u . A value of $\text{category_entropy}(u)$ close to zero indicates strong concentration on a few genres. The interaction count, $\text{interaction_count}(u)$, is the total number of interactions (feedback records) of user u . We define the active user set U_{active} as the users whose $\text{interaction_count}(u)$ exceeds the median over all users, and classify as $G_{\text{Specialist}}$ those users in U_{active} whose $\text{category_entropy}(u)$ falls within the bottom 10%.

We use different cutoffs because the groups capture different behavioral dimensions. $G_{\text{Specialist}}$ is defined as a highly concentrated preference group among active users, so a stricter 10% entropy cutoff is used. G_{Niche} captures low-popularity consumption as a broader behavioral pattern. Thus, the bottom 20% cutoff provides a more stable group size while preserving the niche signal.

The reference group G_{Majority} is defined by the set of users who do not belong to any of the above groups.

2.2 Collapse of SOTA Unlearning Methods

Evaluation Metrics. All ranking metrics are evaluated at $K = 20$. We define $\text{NDCG}_{\text{Overall}}$ as the mean $\text{NDCG}@20$ over all test users, and $\text{NDCG}_{\text{Universal}}$ as the minimum of the group-wise mean $\text{NDCG}@20$ over the four user groups defined in Section 2.1. We define NDCG_{Gap} as the difference between the maximum and minimum group-wise mean $\text{NDCG}@20$. Item Coverage is the fraction of catalog items that appear at least once in the top-20 recommendations across all test users.

Existing approximate unlearning methods exhibit severe performance collapse in the evaluated GNN-based recommendation models when user deletion targets the bridge-like group G_{Niche} . We employ three recommendation models, NGCF [10], GAT [9], and BPR-MF [7], to cover both GNN-based and non-GNN collaborative filtering. We compare three unlearning approaches, Retrain, NegGrad, and SISA, under the 10% user-deletion setting targeting G_{Niche} . The results are summarized in Table 1.

Table 1: Performance collapse of SOTA unlearning methods after 10% user deletion targeting G_{Niche}

Model	Method	$\text{NDCG}_{\text{Overall}} \uparrow$	$\text{NDCG}_{\text{Universal}} \uparrow$	$\text{NDCG}_{\text{Gap}} \downarrow$	Item Coverage $\uparrow$	Time (s) $\downarrow$
NGCF	Pre-unlearning	0.0845	0.0545	0.0351	0.6522	1024.9
	Retrain	0.0892	0.0564	0.0397	0.5796	1114.8
	NegGrad	0.0613 (−31.3%)	0.0244 (−56.7%)	0.0461	0.0653 (−88.7%)	20.0
	SISA	0.0590 (−33.9%)	0.0225 (−60.1%)	0.0470	0.1055 (−81.8%)	870.8
GAT	Pre-unlearning	0.0945	0.0474	0.0528	0.5764	931.3
	Retrain	0.0943	0.0552	0.0470	0.6117	1025.6
	NegGrad	0.0029 (−96.9%)	0.0009 (−98.4%)	0.0052 [†]	0.0159 (−97.4%)	18.6
	SISA	0.0003 (−99.7%)	0.0002 (−99.6%)	0.0002 [†]	0.1543 (−74.8%)	733.3
BPR-MF	Pre-unlearning	0.0979	0.0514	0.0529	0.5192	648.9
	Retrain	0.0964	0.0521	0.0520	0.6249	672.6
	NegGrad	0.0979 (+1.6%)	0.0516 (−1.0%)	0.0533	0.5324 (−14.8%)	8.4
	SISA	0.0724 (−24.9%)	0.0375 (−28.0%)	0.0435	0.4223 (−32.4%)	640.3

Pre-unlearning is shown for reference and excluded from the boldface comparison. [†] denotes an artificially small NDCG Gap caused by near-zero utility collapse; daggered values are excluded from the boldface comparison.

In GAT, NegGrad and SISA cause near-total performance collapse, reducing $\text{NDCG}_{\text{Overall}}$ by 96.9% and 99.7% relative to Retrain, respectively. For NGCF, applying NegGrad also sharply reduces Item Coverage. By contrast, BPR-MF maintains its overall performance under NegGrad, indicating that the severe NegGrad collapse in NGCF and GAT is associated with graph-dependent propagation in this setting. Under Retrain, BPR-MF shows a larger NDCG_{Gap} than NGCF, indicating that switching from NGCF to BPR-MF does not resolve group-wise disparity under the same unlearning procedure. In GAT, NegGrad and SISA produce artificially small NDCG_{Gap} values while $\text{NDCG}_{\text{Universal}}$ collapses to 0.0009 and 0.0002, respectively. These small gaps are therefore artifacts of near-zero utility collapse rather than genuine improvements in group-wise fairness.

3 Ada-Comp to Ease the Bridge-Collapse Dilemma

3.1 Structural Evidence for the Dilemma

To examine the basis of the bridge-collapse dilemma, we analyze two forms of evidence. Evidence 1 examines recommendation performance in Table 2 to identify whether group-wise disparity already appears before unlearning. Evidence 2 analyzes the structural statistics in Tables 3 and 4 to verify whether G_{Niche} plays a bridge-like role in the user–item graph. In Evidence 1, the NDCG of $G_{\text{Specialist}}$ is approximately 39–53% lower than that of G_{Majority} across the three models, suggesting that $G_{\text{Specialist}}$ is vulnerable at the initial recommendation stage. Together with Evidence 2, we interpret this pattern as a structural bottleneck in graph propagation, in which $G_{\text{Specialist}}$ depends on bridge-like users such as G_{Niche} . BPR-MF achieves the highest $\text{NDCG}_{\text{Overall}}$, but also exhibits the largest NDCG_{Gap} and the lowest Item Coverage. In contrast, NGCF achieves the smallest NDCG_{Gap} and the highest Item Coverage among the three models, motivating our focus on preserving the GNN architecture while addressing structural vulnerabilities during unlearning.

To test the noisy-bridge hypothesis, we (i) examine initial group-wise disparity, (ii) compare the centrality of the four user groups to assess the bridge-like role of G_{Niche} , and (iii) analyze the distribution of connection strength between $G_{\text{Specialist}}$ and the other groups to characterize its structural dependence on G_{Niche} .

Table 3 reports centrality statistics by user group, where n denotes the number of unique users in each group. This provides a quantitative comparison of bridge-like roles across the four user groups

Table 2: Initial disparity across recommendation models

Model	NDCG _{Overall} ↑	NDCG _{G_{Niche}} ↑	NDCG _{G_{Sparse}} ↑	NDCG _{G_{Specialist}} ↑	NDCG _{G_{Majority}} ↑	NDCG _{Universal} ↑	NDCG _{Gap} ↓	Item Coverage ↑	Time (s) ↓
NGCF	0.0845	0.0730	0.0827	0.0545 (-39.2%)	0.0896	0.0545	0.0351	0.6522	1024.9
GAT	0.0945	0.0823	0.0920	0.0474 (-52.7%)	0.1002	0.0474	0.0528	0.5764	931.3
BPR-MF	0.0979	0.0829	0.0931	0.0514 (-50.8%)	0.1044	0.0514	0.0529	0.5192	648.9

Table 3: Centrality statistics by user group. Ratios are reported as *Ratio to Majority*

Group	n	Median Centrality	Std	Ratio to Majority
G_{Majority}	3745	5.59×10^{-6}	6.96×10^{-5}	1.00
G_{Niche}	947	4.71×10^{-5}	1.05×10^{-3}	8.43
G_{Sparse}	1208	4.11×10^{-5}	4.34×10^{-4}	7.35
$G_{\text{Specialist}}$	140	1.87×10^{-5}	2.99×10^{-5}	3.35

Table 4: Connection strength statistics by neighbor group. Ratios are reported as *Ratio to Majority*

Group	n	Mean Strength	Std	Ratio to Majority
G_{Majority}	3627	13.35	13.29	1.00
$G_{\text{Specialist}}$	299	19.69	13.69	1.47
G_{Sparse}	918	23.30	18.10	1.74
G_{Niche}	1008	27.69	27.98	2.07

in the user-item graph. G_{Niche} and G_{Sparse} exhibit higher centrality than G_{Majority} . Although $G_{\text{Specialist}}$ also has 3.35 times the centrality of G_{Majority} , its magnitude remains lower than that of the bridge-like groups.

Table 4 complements this analysis by examining the structural dependence of $G_{\text{Specialist}}$ through connection strength. Here, n denotes the number of ego-neighbor pairs for each neighbor group, and the same neighbor may be counted for multiple ego users. On the user-item graph constructed from the training data, we sample users in $G_{\text{Specialist}}$ as ego nodes and compute the connection strength between an ego and each 2-hop neighbor u as

$$S(\text{Ego}, u) = |I_{\text{Ego}} \cap I_u| \quad (4)$$

where I_{Ego} and I_u denote the sets of interacted items. The mean connection strength is higher for G_{Niche} and G_{Sparse} than for G_{Majority} . $G_{\text{Specialist}}$ users are more strongly connected to bridge-like groups than to the reference group, supporting their structural dependence on these groups from a relational perspective.

3.2 Empirical Analysis

To test whether the noisy-bridge hypothesis can guide mitigation of the *bridge-collapse dilemma*, we conduct three empirical analyses. First, we examine whether the initial vulnerability of $G_{\text{Specialist}}$ is related to structural dependence on bridge-like groups. Next, we use a controlled deletion analysis to examine whether the effect of deleting G_{Niche} is associated with its bridge-like role. Finally, we vary the cutoff for defining G_{Niche} to examine whether the bridge-like structural pattern remains consistent.

Table 5 demonstrates that the initial disparity of $G_{\text{Specialist}}$ observed in Section 3.1 arises not merely from sparsity but from a structural bottleneck. If sparsity were the sole cause, deleting users from the other groups would not improve the performance of the remaining $G_{\text{Specialist}}$ users after Retrain. However, as shown in (B)–(D), deleting 10% of users from the other groups and then retraining changes

Table 5: NDCG_{G_{Specialist}} before unlearning and after Retrain under 10% user deletion

Deletion Target	Method	GAT		NGCF	
		NDCG _{G_{Specialist}}	Δ	NDCG _{G_{Specialist}}	Δ
(A) No deletion	Pre-unlearning	0.0474	–	0.0545	–
(B) G_{Niche} (10%)	Retrain	0.0552	+0.0078 (16.5% ↑)	0.0564	+0.0019 (3.5% ↑)
(C) G_{Sparse} (10%)	Retrain	0.0512	+0.0038 (8.0% ↑)	0.0509	-0.0036 (6.6% ↓)
(D) G_{Majority} (10%)	Retrain	0.0538	+0.0064 (13.5% ↑)	0.0572	+0.0027 (5.0% ↑)
(E) $G_{\text{Specialist}}$ (10%)	Retrain	0.0449	-0.0025 (5.3% ↓)	0.0536	-0.0009 (1.7% ↓)

For each model, Δ denotes the absolute difference from its pre-unlearning value.

Table 6: Controlled deletion analysis with NGCF after Retrain under a fixed deletion budget of 120 users. Spec. Drop denotes the relative decrease in the Specialist 2-hop neighborhood

Deletion target	#Users	NDCG _{Overall} ($\Delta\%$)	NDCG _{Universal} ($\Delta\%$)	NDCG _{Gap} ($\Delta\%$)	Spec. Drop	Ratio to Random
No deletion	0	0.0845	0.0545	0.0351	0.00%	–
G_{Niche}	120	0.0817 (–3.3%)	0.0499 (–8.4%)	0.0390 (+11.1%)	7.77%	1.99
Random	120	0.0844 (–0.1%)	0.0516 (–5.3%)	0.0380 (+8.3%)	3.91%	1.00
Degree-matched	120	0.0859 (+1.7%)	0.0548 (+0.6%)	0.0362 (+3.1%)	2.67%	0.68
High-centrality	120	0.0853 (+0.9%)	0.0496 (–9.0%)	0.0408 (+16.2%)	2.41%	0.62

$G_{\text{Specialist}}$ performance in both models. In GAT, the improvement reaches 16.5%. For NGCF, deleting 10% of users from G_{Niche} and then retraining improves $G_{\text{Specialist}}$ performance by 3.5%. Across both evaluated GNN models, $G_{\text{Specialist}}$ performance increases after G_{Niche} deletion and Retrain, showing a consistent directional pattern. Table 6 reports a controlled deletion analysis examining whether the effects of deleting G_{Niche} are associated with its bridge-like role. All protocols remove 120 users, but differ in how deleted users are selected. The main case removes G_{Niche} users. The controls remove random, degree-matched, or high-centrality users to account for generic deletion effects, node degree, and structural centrality, respectively. Relative to the no-deletion baseline, G_{Niche} deletion reduces NDCG_{Overall} by 3.3% and NDCG_{Universal} by 8.4%, while increasing NDCG_{Gap} by 11.1%. Compared with random and degree-matched deletion, G_{Niche} deletion also yields lower NDCG_{Overall}, lower NDCG_{Universal}, and a larger NDCG_{Gap}. These results show that the observed degradation is not explained by generic deletion effects or node degree alone.

The Spec. Drop column provides a graph-side view of the deletion effect by measuring the relative decrease in the two-hop neighborhood of $G_{\text{Specialist}}$. G_{Niche} deletion causes the largest drop in the two-hop neighborhood of $G_{\text{Specialist}}$ among all deletion protocols, suggesting that niche users mediate key connectivity for vulnerable users. Although high-centrality deletion yields the largest NDCG_{Gap}, G_{Niche} deletion causes by far the largest loss of connectivity around $G_{\text{Specialist}}$. This supports the bridge-like role of G_{Niche} and shows that the resulting bridge-removal perturbation cannot be explained by centrality alone.

We test whether the bridge-like structural pattern underlying the noisy-bridge hypothesis depends on the cutoff used to define G_{Niche} . This analysis examines whether the structural pattern remains stable under cutoff changes.

Table 7 shows that the bridge-like structural pattern is consistent across 10%, 20%, and 30% cutoffs. In all cases, G_{Niche} has higher

Table 7: Robustness of the bridge-like structural pattern under different G_{Niche} percentile cutoffs with NGCF. Ratios are computed against G_{Majority}

Cutoff	Betw. Ratio	Strength Ratio	NDCG _{Overall}	NDCG _{Universal}
10%	9.67	2.02	0.0859	0.0527
20%	14.61	2.23	0.0858	0.0540
30%	11.85	2.24	0.0856	0.0537

centrality than G_{Majority} and stronger connections to $G_{\text{Specialist}}$. We use the 20% cutoff as the main setting, consistent with the group definition in Section 2.1.

Taken together, these three analyses show that G_{Niche} is a bridge-like group whose removal weakens connectivity around $G_{\text{Specialist}}$ and amplifies group-wise disparity.

3.3 Adaptive Compensation Strategy

SOTA unlearning methods can undergo severe performance collapse and amplify group-wise disparity in the evaluated GNN models. To ease these limitations, we propose a two-stage framework illustrated in Figure 1.

In the first Retrain stage, we retrain the model from scratch on data from which the target interactions have been removed, yielding the unlearned model denoted Retrain. Table 9 shows that Retrain is stable in terms of overall performance but can retain group-wise disparity. Removing a bridge-like group induces a bridge-removal perturbation and weakens connectivity for structurally dependent users. These structural changes can sustain performance disparities across user groups even after Retrain.

In the second stage, we use the post-unlearning group-wise performance to identify the lowest-performing group. We then apply targeted fine-tuning to the group with the lowest post-unlearning NDCG@K, focusing compensation on its users.

We compute NDCG@K and designate the group with the lowest performance as the affected group. We then continue training Retrain only on this group’s interactions: we construct an *affected-group* dataset from those interactions and fine-tune Retrain with a learning rate of 5×10^{-5} . To account for deletion-induced structural perturbation while limiting overfitting, we run 5 epochs when G_{Niche} is deleted and only 1 epoch when G_{Majority} is deleted.

4 Experimental Results

4.1 Experimental Setup and Evaluation Metrics

We use MovieLens-1M with a time-based leave-10-out split. The evaluated models are BPR-MF, NGCF, and GAT, all with 64-dimensional embeddings. NGCF and GAT use three GNN layers. We evaluate Retrain, NegGrad, SISA, and Ada-Comp at $K = 20$, using a learning rate of 5×10^{-5} for Ada-Comp with candidate fine-tuning epochs $\{1, 3, 5\}$. Unless otherwise stated, we delete 10% of the users from the target group.

4.2 Exploratory Data Analysis

To characterize the four user groups, we analyze 6,040 categorized users, including 947 users in G_{Niche} , 1,208 in G_{Sparse} , 140 in $G_{\text{Specialist}}$,

Table 8: Summary of evaluation metrics

Metric	Definition
NDCG _{Overall}	Mean NDCG@K over all test users.
NDCG _{Universal}	Minimum of the group-wise mean NDCG@K over the four user groups defined in Sec. 2.1.
NDCG _{Gap}	Difference between the maximum and minimum group-wise mean NDCG@K over the four user groups.
Item Coverage	Fraction of catalog items that appear at least once in the top-K recommendations across all users.

Table 9: Comparison of Retrain and Ada-Comp after 10% user deletion targeting G_{Niche}

Model	Method	NDCG _{Overall}	NDCG _{Universal}	NDCG _{Gap}	Item Coverage	Time (s)
NGCF	Retrain	0.0892	0.0564	0.0397	0.5796	1114.8
	Ada-Comp	0.0851	0.0554	0.0356	0.6311	1120.0
GAT	Retrain	0.0943	0.0552	0.0470	0.6117	1025.6
	Ada-Comp	0.0933	0.0524	0.0484	0.6120	1030.5
BPR-MF	Retrain	0.0964	0.0521	0.0520	0.6249	672.6
	Ada-Comp	0.0965	0.0520	0.0523	0.6247	675.9

Table 10: Comparison of Retrain and Ada-Comp after 10% user deletion targeting G_{Majority}

Model	Method	NDCG _{Overall}	NDCG _{Universal}	NDCG _{Gap}	Item Coverage	Time (s)
NGCF	Retrain	0.0876	0.0572	0.0337	0.6160	1010.0
	Ada-Comp	0.0858	0.0563	0.0333	0.6487	1015.0
GAT	Retrain	0.0905	0.0538	0.0391	0.6425	920.4
	Ada-Comp	0.0913	0.0551	0.0386	0.6387	924.9
BPR-MF	Retrain	0.0926	0.0522	0.0432	0.7291	659.6
	Ada-Comp	0.0926	0.0522	0.0432	0.7291	662.9

and 3,745 in G_{Majority} . Figure 2 compares each group with the remaining users (Others) in terms of activity density, consumed-item popularity, and genre entropy.

Figure 2 shows that each group reflects its intended behavioral pattern. G_{Sparse} is concentrated near low activity density, G_{Niche} consumes less popular items than Others, and $G_{\text{Specialist}}$ shows lower genre entropy, indicating more concentrated preferences. These distributions confirm that the resulting segments exhibit their intended behavioral patterns.

4.3 Comparison of Ada-Comp vs. Retrain

Table 9 compares Retrain and Ada-Comp after 10% user deletion targeting G_{Niche} . Ada-Comp is designed as a post-retraining adjustment rather than a replacement for Retrain. For NGCF, Ada-Comp slightly reduces overall accuracy but improves item coverage and lowers NDCG_{Gap}, showing that targeted fine-tuning can reduce group-wise disparity and recover catalog coverage after a bridge-removal perturbation. For GAT and BPR-MF, the changes are marginal, indicating that the benefit of Ada-Comp is most visible in the NGCF setting.

As shown in Table 10, when G_{Majority} is deleted, Ada-Comp is trained for $E = 1$ epoch and produces results close to Retrain. In NGCF, it improves Item Coverage from 0.6160 to 0.6487, while differences in the other metrics remain small, indicating that Ada-Comp provides limited additional benefit under G_{Majority} deletion.

4.4 User Deletion Intensity Experiment

Positive values indicate that Ada-Comp achieves higher NDCG_{Overall} or Item Coverage than Retrain, whereas negative values for NDCG_{Gap}

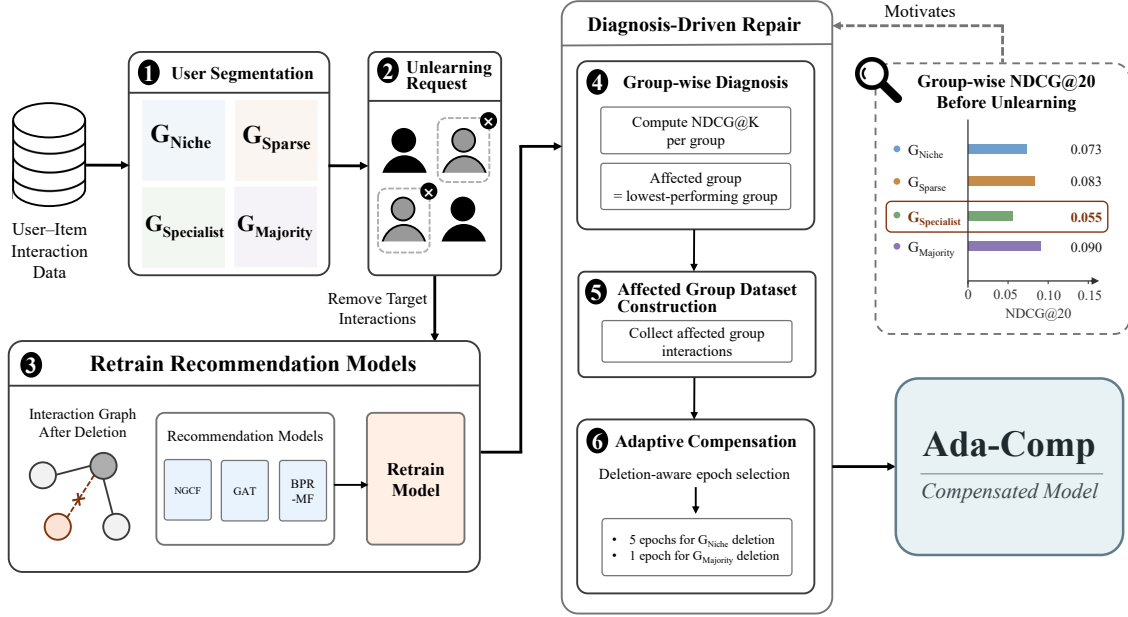


Figure 1: Overview of Ada-Comp.

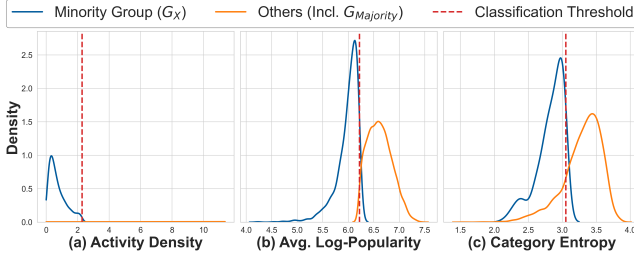


Figure 2: Behavioral profiles of each user group and the remaining users excluding that group.

Table 11: Relative performance difference between Ada-Comp and Retrain across user deletion intensities (%)

Intensity	NGCF (%)			GAT (%)		
	Overall	Gap	Coverage	Overall	Gap	Coverage
1%	+0.23	+0.83	-1.33	-0.31	-1.58	-0.31
5%	-1.81	+1.99	+0.88	-0.21	-1.61	+2.17
10%	-2.15	-1.72	+5.10	+0.53	+2.52	+0.46
20%	+0.46	-0.45	+5.42	-0.21	+0.37	+0.40
30%	-0.70	-2.65	+2.01	-0.88	+0.91	+1.63

indicate reduced group-wise disparity. The results should be interpreted as a sensitivity analysis rather than as evidence of a fixed collapse threshold.

For NGCF, the relative difference in $NDCG_{Overall}$ remains within $\pm 2.15\%$ across all intensities, showing that the overall accuracy gap between Ada-Comp and Retrain does not increase with user deletion intensity. At the 20% and 30% intensities, the difference remains within 1%. The relative difference in $NDCG_{Gap}$ remains within 2.65%, with Ada-Comp reducing group-wise disparity at the 10%, 20%, and 30% intensities. Item Coverage increases by more than 5% at the 10% and 20% intensities, showing that Ada-Comp

Table 12: Comparison of Retrain and Ada-Comp on LastFM-2K after 10% user deletion targeting G_{Niche}

Model	Method	$NDCG_{Overall}$	$NDCG_{Universal}$	$NDCG_{Gap}$	Item Coverage	Time (s)
LightGCN	Retrain	0.1401	0.0680	0.0250	0.2794	105.2
	Ada-Comp	0.1465	0.0829	0.0129	0.2983	112.4

HetRec 2011 LastFM-2K.

expands catalog coverage while maintaining overall accuracy comparable to Retrain.

For GAT, the relative difference in $NDCG_{Overall}$ remains within $\pm 1\%$ across all intensities, showing that Ada-Comp maintains overall accuracy comparable to Retrain across the evaluated user deletion intensities. The relative difference in $NDCG_{Gap}$ remains within approximately 2.5%, with Ada-Comp reducing group-wise disparity at the 1% and 5% intensities but increasing it at the 10%, 20%, and 30% intensities. Item Coverage improves at all intensities except 1%, with the largest gain observed at 5%.

4.5 Cross-Dataset Validation

To examine whether the effectiveness of targeted post-retraining compensation extends beyond MovieLens-1M, we additionally compare Retrain and Ada-Comp on LastFM-2K using LightGCN. As shown in Table 12, Ada-Comp improves $NDCG_{Overall}$ by 4.6% and $NDCG_{Universal}$ by 21.9%, while reducing $NDCG_{Gap}$ by 48.4%. It also improves Item Coverage by 6.8%, demonstrating that Ada-Comp improves overall accuracy, worst-group utility, group-wise balance, and catalog coverage on LastFM-2K.

5 Conclusion

Deleting bridge-like users induces bridge-removal perturbations, under which user-unlearning methods can undergo severe performance collapse in GNN-based recommender systems. The resulting

bridge-collapse pattern can substantially degrade utility and amplify group-wise disparity. We identify G_{Niche} as a bridge-like group whose deletion weakens connectivity around structurally dependent users and increases group-wise disparity in NGCF. We propose Ada-Comp, a post-retraining method that identifies the group with the lowest post-unlearning NDCG@ K and applies targeted compensation to that group. For NGCF after G_{Niche} deletion, Ada-Comp improves Item Coverage by 8.9% and reduces NDCG_{Gap} by 10.3% relative to Retrain, while on LastFM-2K it improves all reported utility, disparity, and coverage metrics. These findings highlight the bridge-collapse dilemma and demonstrate the value of diagnosis-driven post-retraining compensation for mitigating its group-wise effects.

Acknowledgments

This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (RS-2025-25396400), and by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2024-00345398).

References

- [1] Lucas Bourtole, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. 2020. Machine Unlearning. arXiv:1912.03817 [cs.CR] <https://arxiv.org/abs/1912.03817>
- [2] Yushun Dong, Binchi Zhang, Zhenyu Lei, Na Zou, and Jundong Li. 2024. IDEA: A Flexible Framework of Certified Unlearning for Graph Neural Networks. arXiv:2407.19398 [cs.LG] <https://arxiv.org/abs/2407.19398>
- [3] GDPR.eu. 2026. Art. 17 GDPR: Right to erasure ('right to be forgotten'). <https://gdpr.eu/article-17-right-to-be-forgotten/> Accessed: 2026-04-30.
- [4] Yangqin Jiang, Chao Huang, and Lianghao Xia. 2023. Adaptive Graph Contrastive Learning for Recommendation. arXiv:2305.10837 [cs.IR] <https://arxiv.org/abs/2305.10837>
- [5] Zhixun Li, Yushun Dong, Qiang Liu, and Jeffrey Xu Yu. 2024. Rethinking Fair Graph Neural Networks from Re-balancing. arXiv:2407.11624 [cs.LG] <https://arxiv.org/abs/2407.11624>
- [6] Renqiang Luo, Huafei Huang, Shuo Yu, Zhuoyang Han, Estrid He, Xiuzhen Zhang, and Feng Xia. 2024. FUGNN: Harmonizing Fairness and Utility in Graph Neural Networks. arXiv:2405.17034 [cs.LG] <https://arxiv.org/abs/2405.17034>
- [7] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian Personalized Ranking from Implicit Feedback. arXiv:1205.2618 [cs.IR] <https://arxiv.org/abs/1205.2618>
- [8] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. 2009. The graph neural network model. *Trans. Neur. Netw.* 20, 1 (Jan. 2009), 61–80. <https://doi.org/10.1109/TNN.2008.2005605>
- [9] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. arXiv:1710.10903 [stat.ML] <https://arxiv.org/abs/1710.10903>
- [10] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '19)*. ACM, 165–174. <http://dx.doi.org/10.1145/3331184.3331267>
- [11] Kun Wu, Jie Shen, Yue Ning, Ting Wang, and Wendy Hui Wang. 2023. Certified Edge Unlearning for Graph Neural Networks (*KDD '23*). Association for Computing Machinery, New York, NY, USA, 2606–2617. <https://doi.org/10.1145/3580305.3599271>
- [12] Kun Wu and Wendy Hui Wang. 2025. Verification of Incomplete Graph Unlearning through Adversarial Perturbations. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (Toronto ON, Canada) (KDD '25)*. Association for Computing Machinery, New York, NY, USA, 3204–3215. <https://doi.org/10.1145/3711896.3737179>
- [13] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L. Hamilton, and Jure Leskovec. 2018. Graph Convolutional Neural Networks for Web-Scale Recommender Systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '18)*. ACM, 974–983. doi:10.1145/3219819.3219890