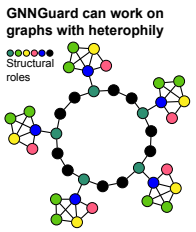
We thank the reviewers for their time and valuable feedback. Overall, we are glad that the reviewers found our GNNGUARD to be *"novel and effective"*, *"model-agnostic"*, *"of practical importance"*, *"does a good job of motivating all of the components"*, has an interesting *"idea of stabilizing training"*, and *"achieves state-of-the-art performance."* Below, we clarify several important points raised by the reviewers. These issues are mainly caused by the omission of certain details due to the limited space. An extra page in the final version will allow us to include the requested details. We believe these clarifications, together with new analyses, resolve all key issues raised.

(1) GNNGUARD can defend graphs with complex patterns beyond homophily. R2, R4, and R5 raise a critical point that GNNGUARD is limited to graphs with homophily. While homophily is a very common assumption in existing research, as nicely pointed out by R1, *"clearly stated by the authors and a reasonable assumption in practice"*. We would like to clarify that GNNGUARD can defend against attacks on graphs with heterophily. In fact, **it is straightforward to use GNNGUARD on graphs with heterophily where connected nodes do not necessarily share similar labels/attributes but share similar roles/positions in the graph (see Figure)**. In response to reviewers, we show how to use GNNGUARD on graphs with structural roles, a prominent type of heterophily. To do this, we replace the cosine similarity (L183, P4) with a graphlet degree similarity [Milenković et al, Cancer Inform.'08], which quantifies structural similarity between nodes in terms of their structural properties, e.g., triangles, betweenness, stars, etc. The graphlet degree similarity is independent of node attributes [Sarajlic et al, Sci. Rep'16] and provides a highly constraining measure of local topology. In the experiment, we synthesized cycle graphs ($N = 1000$, $E = 1600$, $C = 6$) with attached house shapes (see a toy example in Figure), in which node labels are defined by nodes' structural roles [Donnat et al, KDD'18]. We then run GNNGUARD with the following setup: underlying GNN: GIN; attacker: Nettack-Di. We find that GNNGUARD achieves accuracy of 77.5%. In contrast, GNN performance without any defense is only 45%. Further, GNNGUARD outperforms the strongest baseline by 19.2%, which is not surprising as existing GNN defenders cannot defend graphs with heterophily. These new results indicate that GNNGUARD, used in a combination with an appropriate similarity function, can work on graphs with heterophily.



(2) Edge pruning. (2.1) R1 and R2 raise a concern that our ablation does not examine whether edge pruning (Eq. 4-5) is necessary or not. We conduct new experiments showing that edge pruning is a necessary component of GNNGUARD. By removing edge pruning, GNNGUARD's performance drops by 8% on average (see Table) using the setup: dataset: orgb-arxiv; underlying GNN: GIN; attacker: Nettack-Di. Further, edge pruning get even more important when we use GNNGUARD on graphs with heterophily because pruning of adversarial edges has a direct effect on the choice of structural similarity between nodes (e.g., graphlet degree similarity). **(2.2)** R5 requests a clarification of edge pruning across GNN layers. We note that if an edge is pruned in one layer, it will get pruned in all subsequent layers. This is because we quantify similarity s_{uv}^k between u and its neighbor v in each layer (L182-183, P4). Suppose edge e_{uv} is pruned in the (k-1)-th layer, then v is no longer a neighbor of u in the subsequent layers. This means that edge pruning is multi-layer interdependent, and thus GNNGUARD has strong control over the exchange of neural messages in a GNN.

Model	No Defense	**w/o pruning**	w/o memory	GNNGUARD
GCN	0.235	0.350	0.405	0.425
GAT	0.210	0.315	0.475	0.520
GIN	0.315	0.540	0.610	0.640
JK-Net	0.335	0.565	0.625	0.635
GraphSAINT	0.245	0.305	0.360	0.375

(3) GNNGUARD can defend against adaptive attacks. R1 raises an important point regarding the threat of adaptive attacks. We conduct new experiments to evaluate GNNGUARD's ability to defend against adaptive attacks. We develop an Adaptive-Mettack which encourages adversarial edges between nodes with similar representations. In specific, we add the cosine similarity of node-pair (u, v) when calculating the score function $S(u, v)$ [25]. The attacker will select the edge with the highest score as the adversarial edge. On Cora [setup: Cora, GIN, Mettack with 20% budget], we find that the accuracy of GNNGUARD (0.714) surpasses all the baselines including w/o-defense (0.653), GNN-Jaccard (0.679), RobustGCN (0.571), and GNN-SVD (0.683). On Citeseer, GNNGUARD (0.658) also outperforms w/o-defense (0.574), GNN-Jaccard (0.598), RobustGCN (0.583), and GNN-SVD (0.607). The new results show that GNNGUARD can defend GNN models against adaptive attacks. We will carefully discuss the analysis in the final version.

(4) Attacks with varying budgets. R1 raises an important point on examining defense performance with varying budgets. We conducted experiments and can share one of more insightful observations. Non-targeted attacks (e.g., Mettack) are more sensitive to budget amounts than targeted attacks (e.g., Nettack). Mettack causes slight harm within 2% of the budget, but it becomes more harmful when the budget exceeds 10% of the graph size (i.e., number of edges). We find that GNNGUARD consistently outperforms all baselines. We will provide full results in the final version.

(5) Baselines and further clarifications. (5.1) We thank R1 for sharing an interesting survey [Sun et al, arXiv'18], which we have studied extensively and will include it in our final version. We will also carefully discuss other types of attacks (*e.g.*, injecting vicious nodes) and defenders (*e.g.*, defending via VAE). **(5.2)** R5 asks us to justify why ref. [7] is not compared to GNNGUARD. This is a misunderstanding as ref. [7] introduces a GNN attacker, not a GNN defender. Our GNNGUARD is a GNN defender, and thus the comparison is not possible. However, we did use the attacker from [7] (L259-260, P7) to extensively evaluate how successful GNNGUARD's defense is. GNNGUARD achieves strong performance (see Table 1 and Appendix A). We will provide detailed clarifications in our final version.