Deterministic Certification of Graph Neural Networks against Graph Poisoning Attacks with Arbitrary Perturbations

Supplementary Material

A. Proofs

A.1. Proof of Theorem 1

We prove for node classification and it is identical for graph classification.

Recall y_a and y_b are respectively the class with the most vote $\mathbf{n}_{y_a}$ and with the second-most vote $\mathbf{n}_{y_b}$ on predicting the target node v in the subgraphs $\{G_i\}'s$. Hence,

$$\mathbf{n}_{y_a} - \mathbb{I}(y_a > y_b) \geq \mathbf{n}_{y_b} \tag{13}$$

$$\mathbf{n}_{y_b} - \mathbb{I}(y_b > y_c) \geq \mathbf{n}_{y_c}, \forall y_c \in \mathcal{Y} \setminus \{y_a\} \tag{14}$$

where $\mathbb{I}$ is the indicator function, and we pick the class with a smaller index when there exist ties.

Further, on the poisoned classifiers $f'_{[S]}$ with $\theta'_{[S]}$ after the attack, the vote $\mathbf{n}'_{y_a}$ of the class y_a and vote $\mathbf{n}'_{y_c}$ of any other class $y_c \in \mathcal{Y} \setminus \{y_a\}$ satisfy the below relationship:

$$\mathbf{n}'_{y_a} \geq \mathbf{n}_{y_a} - \sum_{i=1}^{T} \mathbb{I}(f_i(G_i)_v \neq f'_i(G_i)_v) \tag{15}$$

$$\mathbf{n}'_{y_c} \leq \mathbf{n}_{y_c} + \sum_{i=1}^{T} \mathbb{I}(f_i(G_i)_v \neq f'_i(G_i)_v) \tag{16}$$

Since $f_{[S]}$ and $f'_{[S]}$ only differ in trained weights, the above expression $\sum_{i=1}^{T} \mathbb{I}(f_i(G_i)_v \neq f'_i(G_i)_v)$ could be replaced by $\sum_{i=1}^{T} \mathbb{I}(\theta_i \neq \theta'_i)$

To ensure the returned label by the voting node classifier $\bar{f}$ does not change, i.e., $\bar{f}(G)_v = \bar{f}'(G)_v, \forall \mathcal{G}'_{tr}$, we must have:

$$\mathbf{n}'_{y_a} \geq \mathbf{n}'_{y_c} + \mathbb{I}(y_a > y_c), \forall y_c \in \mathcal{Y} \setminus \{y_a\} \tag{17}$$

Combining with Eqns 15 and 16, the sufficient condition for Eqn 17 to satisfy is to ensure:

$$\mathbf{n}_{y_a} - \sum_{i=1}^{T} \mathbb{I}(\theta_i \neq \theta'_i) \geq \mathbf{n}_{y_c} + \sum_{i=1}^{T} \mathbb{I}(\theta_i \neq \theta'_i) \tag{18}$$

Or,

$$\mathbf{n}_{y_a} \geq \mathbf{n}_{y_c} + 2\sum_{i=1}^{T} \mathbb{I}(\theta_i \neq \theta'_i) + \mathbb{I}(y_a > y_c). \tag{19}$$

Plugging Eqn 14, we further have this condition:

$$\mathbf{n}_{y_a} \geq \mathbf{n}_{y_b} - \mathbb{I}(y_b > y_c) + 2\sum_{i=1}^{T} \mathbb{I}(\theta_i \neq \theta'_i) + \mathbb{I}(y_a > y_c) \tag{20}$$

We observe that:

$$\mathbb{I}(y_a > y_b) \geq \mathbb{I}(y_a > y_c) - \mathbb{I}(y_b > y_c), \forall y_c \in \mathcal{Y} \setminus \{y_a\} \tag{21}$$

Combining Eqn 21 with Eqn 20, we have:

$$\mathbf{n}_{y_a} \geq \mathbf{n}_{y_b} + 2\sum_{i=1}^{T} \mathbb{I}(\theta_i \neq \theta'_i) + \mathbb{I}(y_a > y_b) \tag{22}$$

Let $M = \lfloor \mathbf{n}_{y_a} - \mathbf{n}_{y_b} - \mathbb{I}(y_a > y_b) \rfloor / 2$, hence

$$\sum_{i=1}^{T} \mathbb{I}(\theta_i \neq \theta'_i) \leq M.$$

A.2. Proof of Theorem 2

To prove Theorem 2, we will first certify the bounded number of altered predictions under (1) edge manipulation, (2) node manipulation and (3) node feature manipulation separately through Theorems 6-8.

Theorem 6. *Assume $\mathcal{G}_{tr}$ is under the edge manipulation $\{\mathcal{E}_+, \mathcal{E}_-\}$, then at most $|\mathcal{E}_+| + |\mathcal{E}_-|$ sub-classifiers trained by our edge-centric subgraph sets are different between $\mathcal{G}'_{[S]}$ and $\mathcal{G}_{[S]}$.*

Proof. Edges of a train graph G in all subgraph sets of $\mathcal{G}_{[S]}$ are disjoint. Hence, when any edge in G is deleted or added by an adversary, only one subgraph set in $\mathcal{G}_{[S]}$ is affected. Further, when any $|\mathcal{E}_+| + |\mathcal{E}_-|$ edges in G are perturbed, there are at most $|\mathcal{E}_+| + |\mathcal{E}_-|$ subgraph set between $\mathcal{G}_{[S]}$ and $\mathcal{G}'_{[S]}$ are different. By training S node/graph sub-classifiers on $\mathcal{G}_{[S]}$ and $\mathcal{G}'_{[S]}$, there are at most $|\mathcal{E}_+| + |\mathcal{E}_-|$ sub-classifiers that have different weights between them. □

Theorem 7. *Assume the training graph set $\mathcal{G}_{tr}$ is under the node manipulation $\{\mathcal{V}_+, \mathcal{E}_{\mathcal{V}_+}, \mathbf{X}'_{\mathcal{V}_+}, \mathcal{V}_-, \mathcal{E}_{\mathcal{V}_-}\}$, then at most $|\mathcal{E}_{\mathcal{V}_+}| + |\mathcal{E}_{\mathcal{V}_-}|$ sub-classifiers trained by our edge-centric subgraph sets are different between $\mathcal{G}'_{[S]}$ and $\mathcal{G}_{[S]}$.*

Theorem 8. *Assume the training graph set $\mathcal{G}_{tr}$ is under the node feature manipulation $\{\mathcal{V}_r, \mathcal{E}_{\mathcal{V}_r}, \mathbf{X}'_{\mathcal{V}_r}\}$, then at most $|\mathcal{E}_{\mathcal{V}_r}|$ sub-classifiers trained by our edge-centric subgraph sets are different between $\mathcal{G}'_{[S]}$ and $\mathcal{G}_{[S]}$.*

Proof. Our proof for the above two theorems is based on the key observation that manipulations on isolated nodes do not participate in the forward calculation of other nodes' representations in GNNs. Take node injection for instance and the proof for other cases are similar. Note that all subgraphs after node injection will contain the newly injected nodes, but they still do not have overlapped edges between each other via the hash mapping. Hence, the edges $E_{\mathcal{V}_+}$ induced by the injected nodes $\mathcal{V}_+$ exist in at most $|E_{\mathcal{V}_+}|$ subgraphs. In other word, the injected nodes $\mathcal{V}_+$ in at least $S - |E_+|$ subgraphs have no edges and are isolated.

Due to the message passing mechanism in GNNs, every node only uses its neighboring nodes' representations to update its own representation. Hence, these subgraphs with the isolated injected nodes, whatever their features $\mathbf{X}'_{\mathcal{V}_+}$ are, would have no influence on other nodes' representation calculation. Therefore, in at least $S - |E_+|$ subgraph sets, the training nodes'/graphs' representations and gradients maintain the same, implying the trained classifier weight to be the same. □

By combining above theorems, we could reach Theorem 2 by simply adding up the bounded number.

A.3. Proof of Theorem 4

Similar to the proof of Theorem 2, to prove Theorem 4, we first certify the bounded number of altered predictions under (1) edge manipulation, (2) node manipulation and (3) node feature manipulation separately through Theorems 9-11.

Theorem 9. *Assume $\mathcal{G}_{tr}$ is under the edge manipulation $\{\mathcal{E}_+, \mathcal{E}_-\}$, then at most $2|\mathcal{E}_+| + 2|\mathcal{E}_-|$ node sub-classifiers trained by our node-centric subgraph sets are different between $\vec{\mathcal{G}}'_{[S]}$ and $\vec{\mathcal{G}}_{[S]}$, and at most $|\mathcal{E}_+| + |\mathcal{E}_-|$ graph sub-classifiers trained by our node-centric subgraph sets are different between $\vec{\mathcal{G}}'_{[S]}$ and $\vec{\mathcal{G}}_{[S]}$.*

Proof. For the node classifier, We simply analyze when an arbitrary edge (u, v) is deleted/added from a train graph $G \in \mathcal{G}_{\text{tr}}$. It is obvious at most two subgraphs $\vec{G}_{i_{u\to v}}$ and $\vec{G}_{i_{v\to u}}$ are perturbed after perturbation, and therefore two subgraph sets are affected. Generalizing this observation to any $|\mathcal{E}_+| + |\mathcal{E}_-|$ edges in G being perturbed, at most $2|\mathcal{E}_+| + 2|\mathcal{E}_-|$ subgraph sets are generated different between $\mathcal{G}_{[S]}$ and $\mathcal{G}'_{[S]}$.

For the graph classifier, we consider the following two cases: i) $i_{u\to v} = i_{v\to u}$. this means u and v are in the same subgraph, hence at most one subgraph's representation is affected; ii) $i_{u\to v} \neq i_{v\to u}$. Due to the removal of other nodes whose subgraph index is not i in every subgraph $\vec{G}_i$, both direct edges would always be removed from $\vec{G}_{i_{u\to v}}$ and $\vec{G}_{i_{v\to u}}$ if exist. Generalizing this observation to any $|\mathcal{E}_+| + |\mathcal{E}_-|$ edges in G being perturbed, at most $|\mathcal{E}_+| + |\mathcal{E}_-|$ subgraph sets are generated different between $\mathcal{G}_{[S]}$ and $\mathcal{G}'_{[S]}$. □

Theorem 10. *Assume a graph G is under the node manipulation $\{\mathcal{V}_+, \mathcal{E}_{\mathcal{V}_+}, \mathbf{X}'_{\mathcal{V}_+}, \mathcal{V}_-, \mathcal{E}_{\mathcal{V}_-}\}$, then at most $|\mathcal{V}_+| + |\mathcal{V}_-|$ node/graph sub-classifiers trained by our node-centric subgraph sets are different between $\vec{\mathcal{G}}'_S$ and $\vec{\mathcal{G}}_S$.*

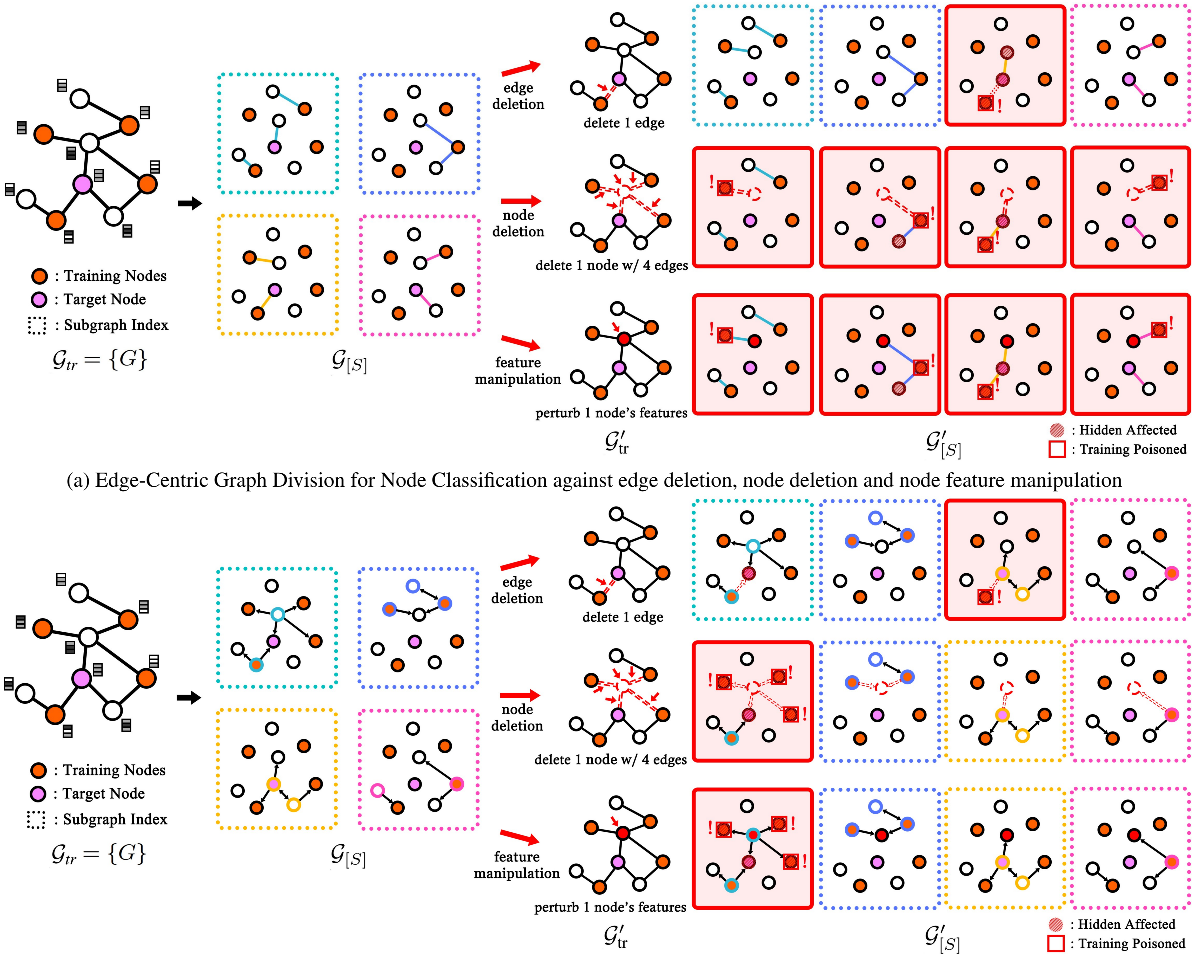
Node Classification	**Ave degree**	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{E}\rvert$	$\lvert\mathcal{C}\rvert$
Cora-ML	5.6	2, 995	8,416	7
Citeseer	2.8	3,327	4,732	6
Pubmed	4.5	19,717	44,338	3
Amazon-C	71.5	13,752	491,722	10
Graph Classification	$\lvert\mathcal{G}\rvert$	$\lvert\mathcal{V}\rvert_{avg}$	$\lvert\mathcal{E}\rvert_{avg}$	$\lvert\mathcal{C}\rvert$
AIDS	2,000	15.7	16.2	2
MUTAG	4,337	30.3	30.8	2
PROTEINS	1,113	39.1	72.8	2
DD	1,178	284.3	715.7	2

Table 5. Datasets and their statistics.

Theorem 11. *Assume a graph G is under the node feature manipulation $\{\mathcal{V}_r, \mathcal{E}_{\mathcal{V}_r}, \mathbf{X}'_{\mathcal{V}_r}\}$, then at most $|\mathcal{V}_r|$ node/graph sub-classifiers trained by our node-centric subgraphs are different between $\vec{\mathcal{G}}'_S$ and $\vec{\mathcal{G}}_S$.*

Proof. Our proof for the above two theorems is based on the key observation that: *in a directed graph, manipulations on nodes with no outgoing edge have no influence on other nodes' representations in GNNs.* For any node $u \in G$, only one subgraph $\vec{G}_{h[\mathtt{str}(u)]\ \mathtt{mod}\ S+1}$ has outgoing edges. Take node injection for instance and the proof for other cases are similar. Note that all subgraphs after node injection will contain newly injected nodes V_+, but they still do not have overlapped nodes with outgoing edges between each other via the hashing mapping. Hence, the injected nodes only have outgoing edges in at most $|V_+|$ subgraphs. Due to the directed message passing mechanism in GNNs, every node only uses its incoming neighboring nodes' representation to update its own representation. Hence, the injected nodes with no outgoing edges, whatever their features $\mathbf{X}'_{\mathcal{V}_+}$ are, would have no influence on other nodes' representation and gradients, including the training nodes', implying at least $S - |V_+|$ subgraphs' training process maintain the same. □

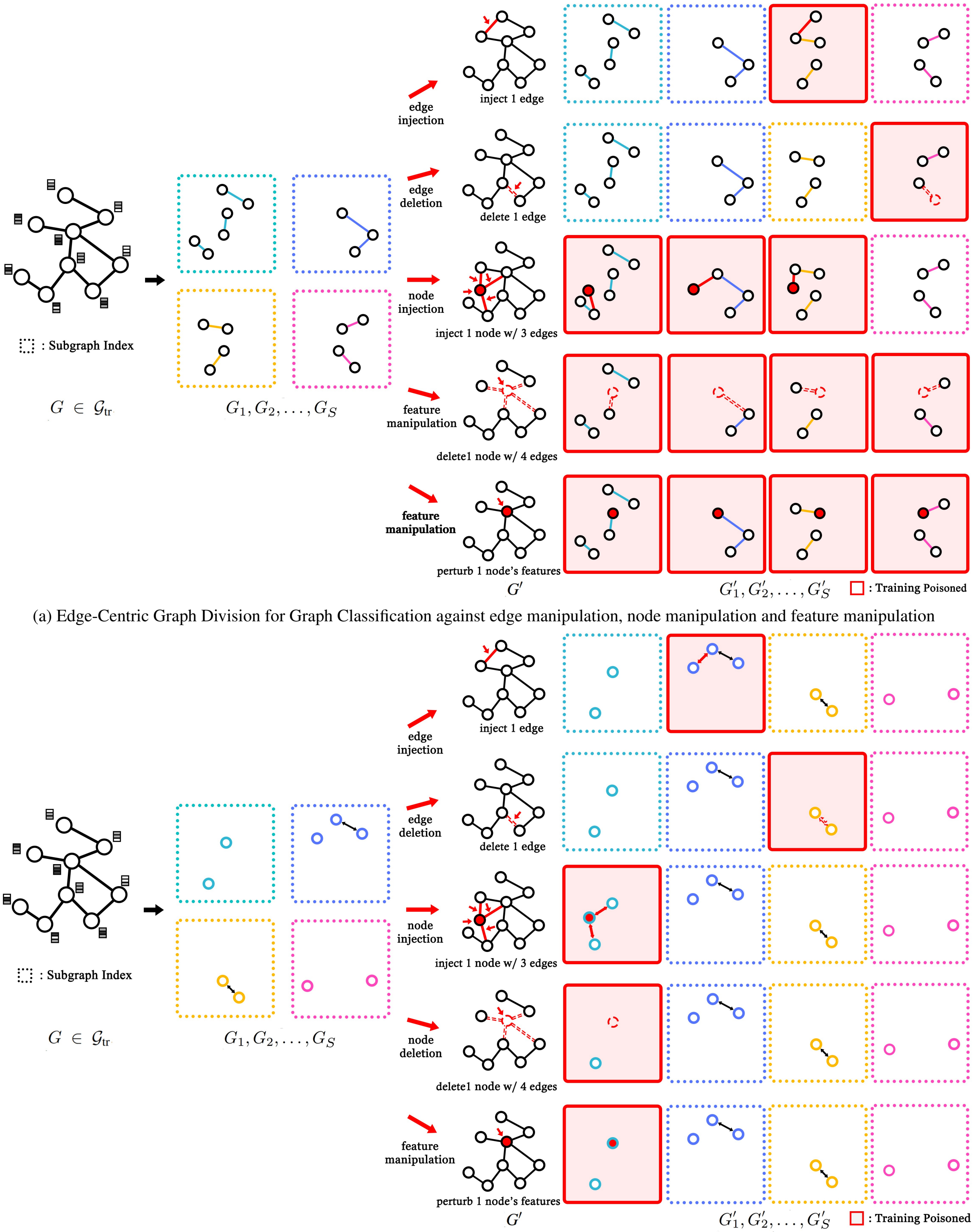
By collaborating above theorems together, we could reach Theorem 4 by simply adding up the bounded number.



(a) Edge-Centric Graph Division for Node Classification against edge deletion, node deletion and node feature manipulation

(b) Node-Centric Graph Division for Node Classification against edge deletion, node deletion and node feature manipulation

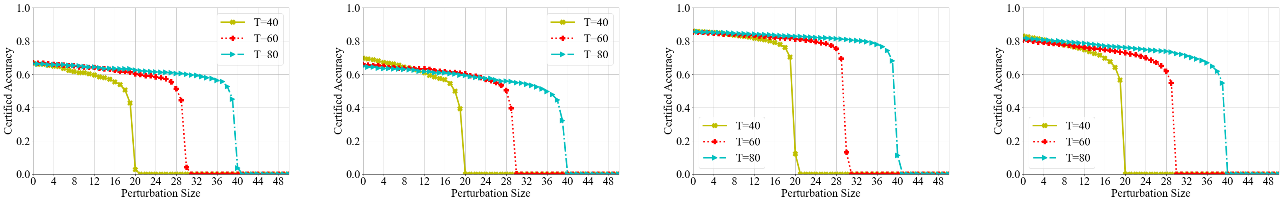
Figure 8. Illustration of our edge-centric and node-centric graph division strategies for node classification against edge deletion, node deletion, and node feature manipulation. **To summarize:** 1 deleted edge affects at most 1 subgraph prediction in both graph division strategies. In contrast, 1 deleted node with, e.g., 3 incident edges can affect at most 3 subgraph predictions with edge-centric graph division, but at most 1 subgraph prediction with node-centric graph division.



(a) Edge-Centric Graph Division for Graph Classification against edge manipulation, node manipulation and feature manipulation

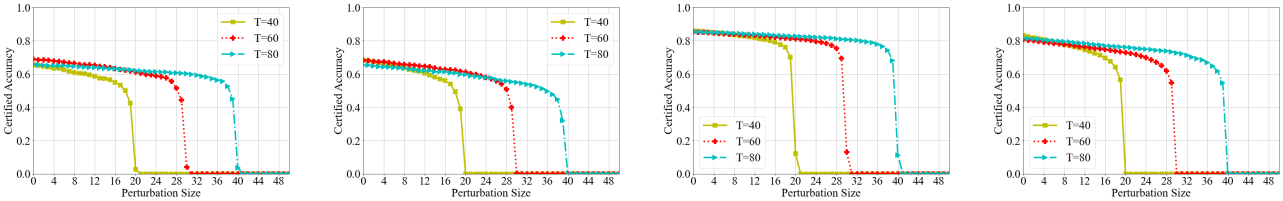
(b) Node-Centric Graph Division for Graph Classification against edge manipulation, node manipulation and feature manipulation

Figure 9. Illustration of our edge-centric and node-centric graph division strategies for graph classification. The conclusion are similar to those for node classification.



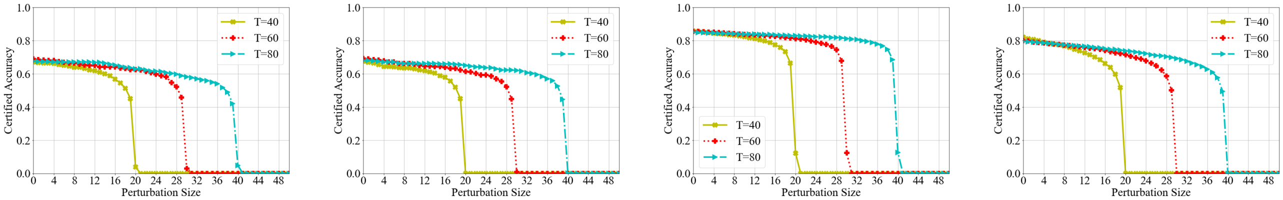
(a) Cora-ML (b) Citeseer (c) Pubmed (d) Amazon-C

Figure 10. Certified node accuracy of our PGNNCert−E with GSAGE w.r.t. the number of subgraphs S.



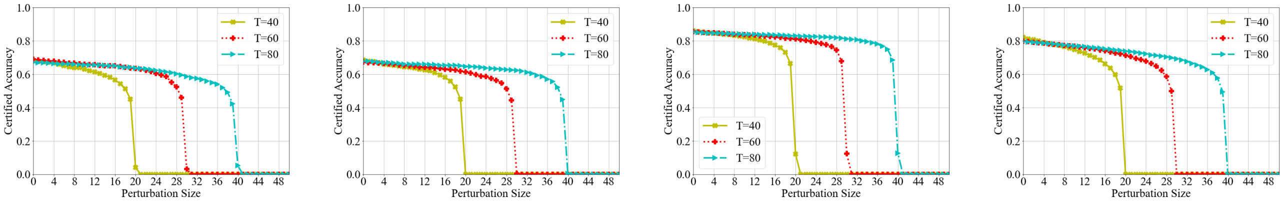
(a) Cora-ML (b) Citeseer (c) Pubmed (d) Amazon-C

Figure 11. Certified node accuracy of our PGNNCert−N with GSAGE w.r.t. the number of subgraphs S.



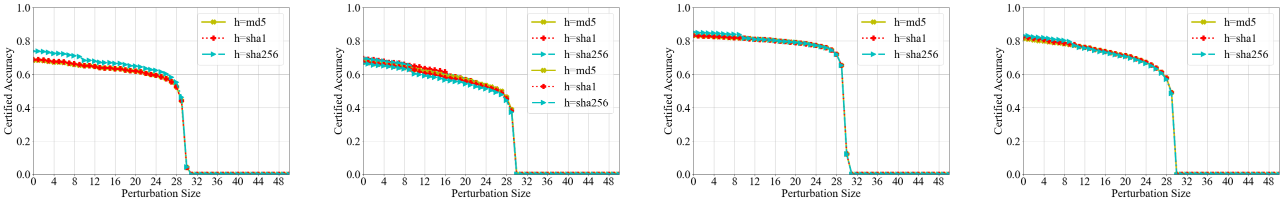
(a) Cora-ML (b) Citeseer (c) Pubmed (d) Amazon-C

Figure 12. Certified node accuracy of our PGNNCert−E with GAT w.r.t. the number of subgraphs S.



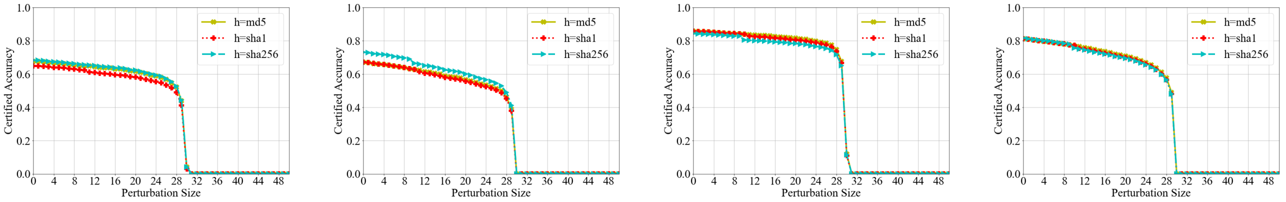
(a) Cora-ML (b) Citeseer (c) Pubmed (d) Amazon-C

Figure 13. Certified node accuracy of our PGNNCert−N with GAT w.r.t. the number of subgraphs S.



(a) Cora-ML (b) Citeseer (c) Pubmed (d) Amazon-C

Figure 14. Certified node accuracy of our PGNNCert−E w.r.t. the hash function h.



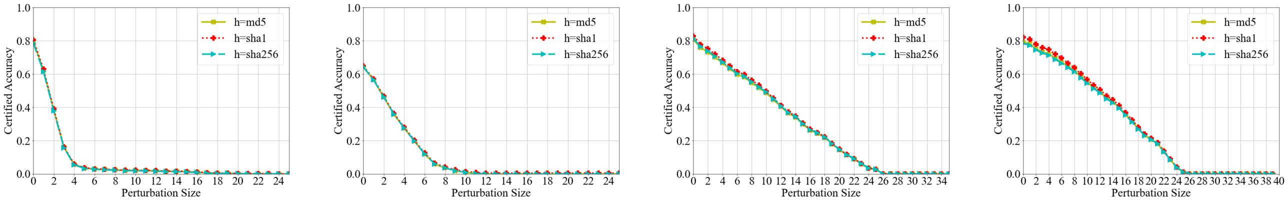
(a) Cora-ML

(b) Citeseer

(c) Pubmed

(d) Amazon-C

Figure 15. Certified node accuracy of our PGNNCert−N w.r.t. the hash function h.



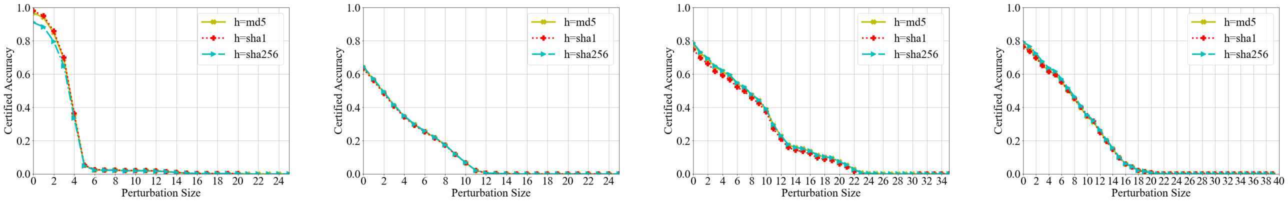
(a) AIDS

(b) MUTAG

(c) PROTEINS

(d) DD

Figure 16. Certified graph accuracy of our PGNNCert−E w.r.t. the hash function h.



(a) AIDS

(b) MUTAG

(c) PROTEINS

(d) DD

Figure 17. Certified graph accuracy of our PGNNCert−N w.r.t. the hash function h.