Inference Attacks Against Graph Neural Networks

Authors: Zhikun Zhang, Min Chen, Michael Backes, Yun Shen, Yang Zhang
Presented by Qianjun Wei

Inference Attacks Against Graph Neural Networks [USENIX Security 22]

- Classification tasks in graph neural networks
  - Node Classification
    - To determine the label of nodes in the graph
      - Such as the gender of a user in a social network
    - What GNNs do:
      - Generate node embeddings
      - Feed them to a classifier to determine the node labels

Inference Attacks Against Graph Neural Networks [USENIX Security 22]

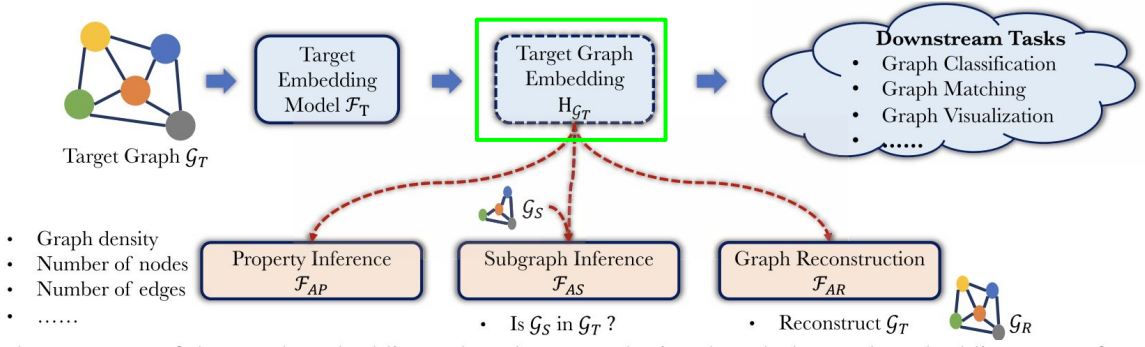
- Classification tasks in graph neural networks
  - Graph Classification
    - To determine the label of the whole graph
      - such as a molecule's solubility or toxicity
    - What can GNNs do:
      - Generate node embeddings
      - Transform all the node embeddings to a whole **graph embedding**
      - Feed **graph embeddings** to a classifier to determine the whole graph labels

Inference Attacks Against Graph Neural Networks [USENIX Security 22]

- Motivation
  - In practice, some **graph embeddings** have been shared with third parties to conduct downstream graph analysis tasks
    - Some companies release their graph embedding systems, together with which they publish some **pretrained graph embeddings** to facilitate the downstream tasks
      - PyTorch BigGraph system
      - DGL-KE system
      - …
  - But **graph embeddings** will leak sensitive structural information of corresponding graph **!**

Inference Attacks Against Graph Neural Networks [USENIX Security 22]

- Inference Attacks on GNNs:
  - Property Inference:
    - The number of nodes or edges, the density
  - Subgraph Inference:
    - Whether a small graph is contained in the target graph
  - Graph reconstruction:
    - Reconstruct a graph have similar statistics as target graph
      - Such as degree distribution, local clustering coefficient, etc.



Inference Attacks Against Graph Neural Networks [USENIX Security 22]

- What the adversary knows：
  - The graph embedding of target graph
    - Access to the graph embedding
  - An auxiliary dataset
    - In the same distribution as target graph
    - Is used for train the attack models in three inference attacks

Inference Attacks Against Graph Neural Networks [USENIX Security 22]

- Property Inference:
  - Attacker uses **auxiliary dataset** to train this inference attack model.
    - Feature extractor: multiple sequential linear layers
    - Multiple parallel prediction layers
  - Evaluation Metrics:
    - Attack accuracy
      - calculates the proportion of graphs being correctly inferred

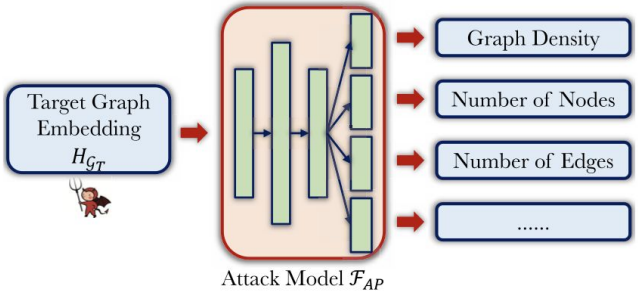


Figure 2: Attack pipeline of the property inference attack. The attack model $\mathcal{F}_{AP}$ is a multi-task classifier, which consists of multiple output layers, each predicts one graph property.

Training Attack Model. Recall that the attack model $\mathcal{F}_{AP}$ is the combination of a feature extractor $\mathcal{E}$ and multiple prediction layers $\mathcal{M}$, we can train the attack model by optimizing the following optimization problem:

$$\min \underset{\mathcal{G}_{aux} \in \mathcal{D}_{aux}}{\mathbb{E}} \left[\sum_{p \in \mathbb{P}} \mathcal{L}\left[\mathcal{M}^p(\mathcal{E}(H_{\mathcal{G}_{aux}})), p\right] \right]$$

where $\mathbb{P}$ is the set of properties that the attackers interested, p is a property in $\mathbb{P}$, $\mathcal{L}$ is the cross-entropy loss. Notice that all properties share the same parameters for $\mathcal{E}$, and use different parameters for $\mathcal{M}^p$.

Inference Attacks Against Graph Neural Networks [USENIX Security 22]

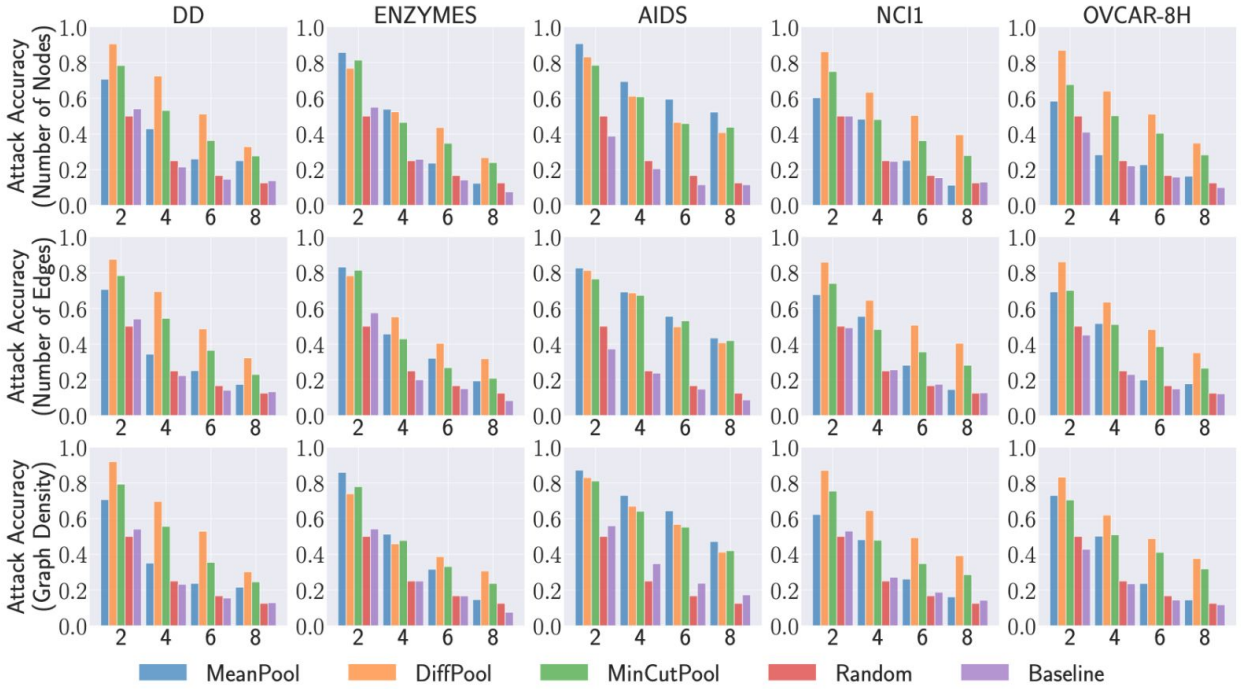


Figure 5: [Higher means better attack performance.] Attack accuracy for property inference. Different columns represent different datasets, and different rows represent different graph properties to be inferred. In each figure, different legends stand for different graph embedding models, different groups stand for different bucketization schemes. The Random and Baseline method represent the random guessing and summarizing auxiliary dataset baseline, respectively.

Inference Attacks Against Graph Neural Networks [USENIX Security 22]

- Subgraph Inference:
  - Attack Goal:
    - Given the target graph embedding and a subgraph of interest, the attack goal is to infer whether subgraph is contained in target graph.
  - Subgraph Inference Attack model
    - Embedding Extractor
      - to transform the subgraph to a subgraph embedding
    - Binary classifier

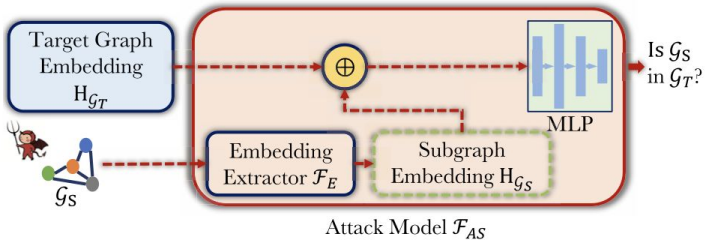


Figure 3: Attack pipeline of the subgraph inference attack. The attack model $\mathcal{F}_{AS}$ has two inputs with different formats, namely target graph embedding and subgraph. The subgraph is transformed to a subgraph embedding by an embedding extractor integrated in the attack model, aggregated with the target embedding, and sent to a binary classifier for prediction.

Inference Attacks Against Graph Neural Networks [USENIX Security 22]

- Subgraph Inference:
  - Constructing Features
    - To generate an binary classification vector for MLP
      - Concatenation
      - Element-wise Diff
      - Euclidean Diff
  - Binary classifier model
    - MLP
    - Cross entropy loss
    - Gradient decent algorithm
  - Evaluation Metrics
    - AUC of binary classification

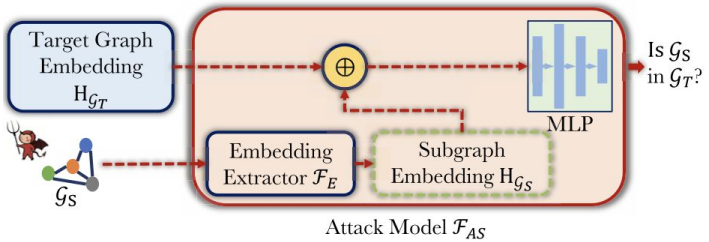


Figure 3: Attack pipeline of the subgraph inference attack. The attack model $\mathcal{F}_{AS}$ has two inputs with different formats, namely target graph embedding and subgraph. The subgraph is transformed to a subgraph embedding by an embedding extractor integrated in the attack model, aggregated with the target embedding, and sent to a binary classifier for prediction.

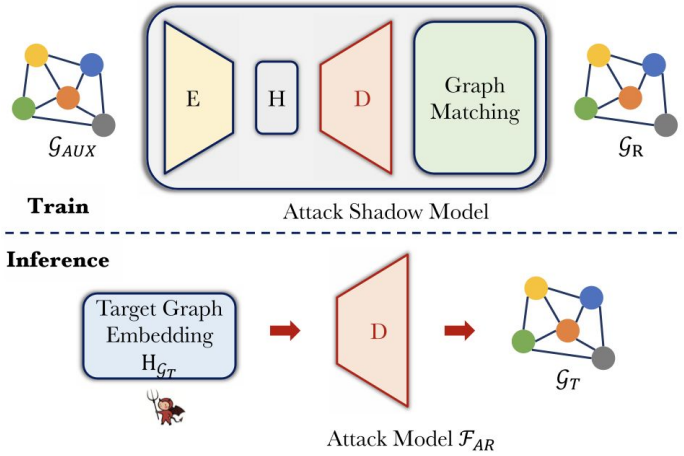
Inference Attacks Against Graph Neural Networks [USENIX Security 22]

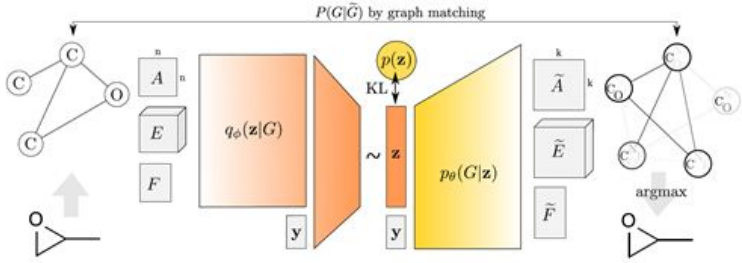
Table 2: Attack AUC for different feature construction methods in subgraph inference attack. The graph embedding model is DiffPool and the graph sampling method is RandomWalk. Due to space limitation, we use Concat, EDist, and EDiff to represent Concatenation, Euclidean Distance, and Element-wise Difference, respectively.

Dataset	0.8			0.6			0.4			0.2		
	Concat	EDist	EDiff	Concat	EDist	EDiff	Concat	EDist	EDiff	Concat	EDist	EDiff
DD	0.53 ± 0.01	0.81 ± 0.06	$\mathbf{0.88 \pm 0.01}$	0.51 ± 0.01	0.79 ± 0.04	$\mathbf{0.87 \pm 0.01}$	0.52 ± 0.01	0.79 ± 0.02	$\mathbf{0.85 \pm 0.01}$	0.50 ± 0.02	0.71 ± 0.08	$\mathbf{0.80 \pm 0.00}$
ENZYMES	0.49 ± 0.02	0.63 ± 0.10	$\mathbf{0.88 \pm 0.03}$	0.52 ± 0.03	0.71 ± 0.10	$\mathbf{0.88 \pm 0.03}$	0.54 ± 0.02	0.56 ± 0.07	$\mathbf{0.86 \pm 0.01}$	0.48 ± 0.02	0.53 ± 0.03	$\mathbf{0.78 \pm 0.01}$
AIDS	0.51 ± 0.01	0.53 ± 0.04	$\mathbf{0.78 \pm 0.04}$	0.55 ± 0.01	0.51 ± 0.02	$\mathbf{0.76 \pm 0.05}$	0.54 ± 0.01	0.51 ± 0.03	$\mathbf{0.73 \pm 0.06}$	0.56 ± 0.02	0.50 ± 0.00	$\mathbf{0.76 \pm 0.05}$
NCI1	0.51 ± 0.00	0.51 ± 0.02	$\mathbf{0.70 \pm 0.06}$	0.49 ± 0.02	0.52 ± 0.01	$\mathbf{0.67 \pm 0.06}$	0.50 ± 0.01	0.51 ± 0.01	$\mathbf{0.64 \pm 0.03}$	0.49 ± 0.01	0.51 ± 0.01	$\mathbf{0.64 \pm 0.00}$
OVCAR-8H	0.54 ± 0.01	0.63 ± 0.12	$\mathbf{0.89 \pm 0.02}$	0.50 ± 0.04	0.69 ± 0.09	$\mathbf{0.88 \pm 0.02}$	0.51 ± 0.03	0.74 ± 0.02	$\mathbf{0.84 \pm 0.01}$	0.54 ± 0.01	0.60 ± 0.13	$\mathbf{0.82 \pm 0.02}$

Inference Attacks Against Graph Neural Networks [USENIX Security 22]

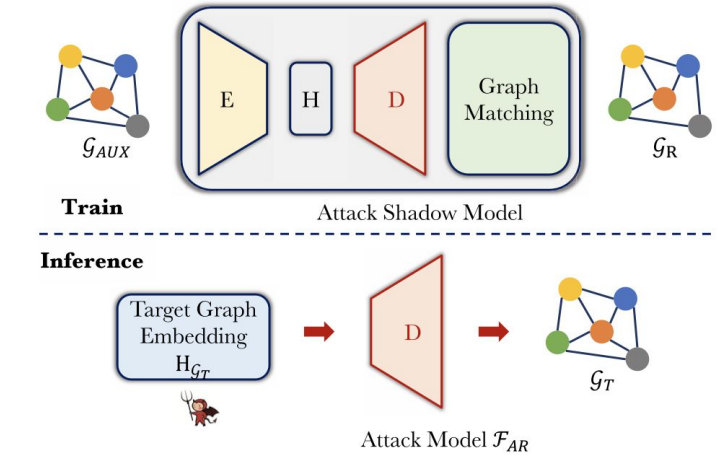
- Graph restruction:
  - Attack Model
    - The attack model is decoder of GraphVAE
    - Auto-encoder model is trained by auxiliary dataset
    - Encoder is GNN model
    - Decoder is MLP

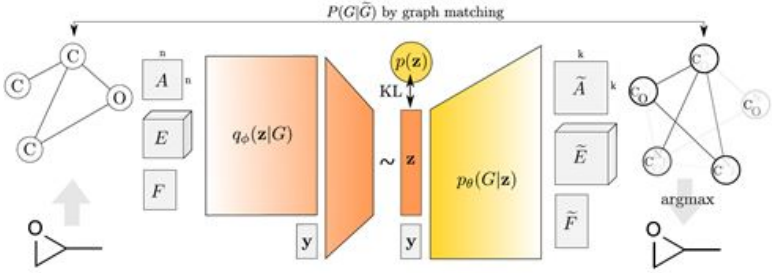




Inference Attacks Against Graph Neural Networks [USENIX Security 22]

- Graph restruction:
  - Disscussion
    - Both the space and time complexity of the graph matching algorithm are O(n4)
    - Graph matching
    - Thus, this attack can be only applied to graphs with **tens** of nodes





Inference Attacks Against Graph Neural Networks [USENIX Security 22]

Table 3: [Higher means better attack performance.] Attack performance of graph reconstruction measured by graph isomorphism.

Dataset	DiffPool	MeanPool	MinCutPool
AIDS	0.875 ± 0.003	0.794 ± 0.003	0.869 ± 0.002
ENZYMES	0.670 ± 0.019	0.653 ± 0.022	0.704 ± 0.012
NCI1	0.752 ± 0.005	0.771 ± 0.010	0.693 ± 0.007

Table 4: [Higher means better attack performance.] Attack performance of graph reconstruction measured by macro-level graph statistics, the similarity of which is measured by cosine similarity.

Dataset	Target Model	Degree Dist.	LCC Dist.	BC Dist.	CC Dist.
AIDS	MeanPool	0.651 ± 0.001	0.999 ± 0.001	0.987 ± 0.001	0.876 ± 0.002
	DiffPool	0.894 ± 0.001	0.999 ± 0.001	0.983 ± 0.001	0.787 ± 0.002
	MinCutPool	0.888 ± 0.003	0.999 ± 0.001	0.983 ± 0.001	0.785 ± 0.006
ENZYMES	MeanPool	0.450 ± 0.070	0.646 ± 0.005	0.959 ± 0.001	0.516 ± 0.037
	DiffPool	0.519 ± 0.007	0.661 ± 0.008	0.958 ± 0.001	0.504 ± 0.005
	MinCutPool	0.467 ± 0.019	0.490 ± 0.009	0.916 ± 0.001	0.414 ± 0.009
NCI1	MeanPool	0.736 ± 0.003	0.999 ± 0.001	0.877 ± 0.001	0.402 ± 0.001
	DiffPool	0.633 ± 0.002	0.999 ± 0.001	0.877 ± 0.001	0.495 ± 0.002
	MinCutPool	0.570 ± 0.002	0.999 ± 0.001	0.877 ± 0.001	0.496 ± 0.001

Inference Attacks Against Graph Neural Networks [USENIX Security 22]

- Conclusion
  - This paper investigate the information leakage of graph embedding.
  - They propose three different attacks to extract information from the target graph given the graph embedding.
    - Graph properties
    - Subgraph inference
    - Graph reconstruction
  - The auxiliary dataset is important.
  - Shortcomings
    - This paper doesn't select good metrics for estimating the effectiveness of the inference attack
    - In the property inference attack, the effectiveness is strongly correlated with the bucket size.

Thank you