Model Inversion Attack: From Images to Graphs

Presenter: Zhanke Zhou

2023. 02. 07

Outline

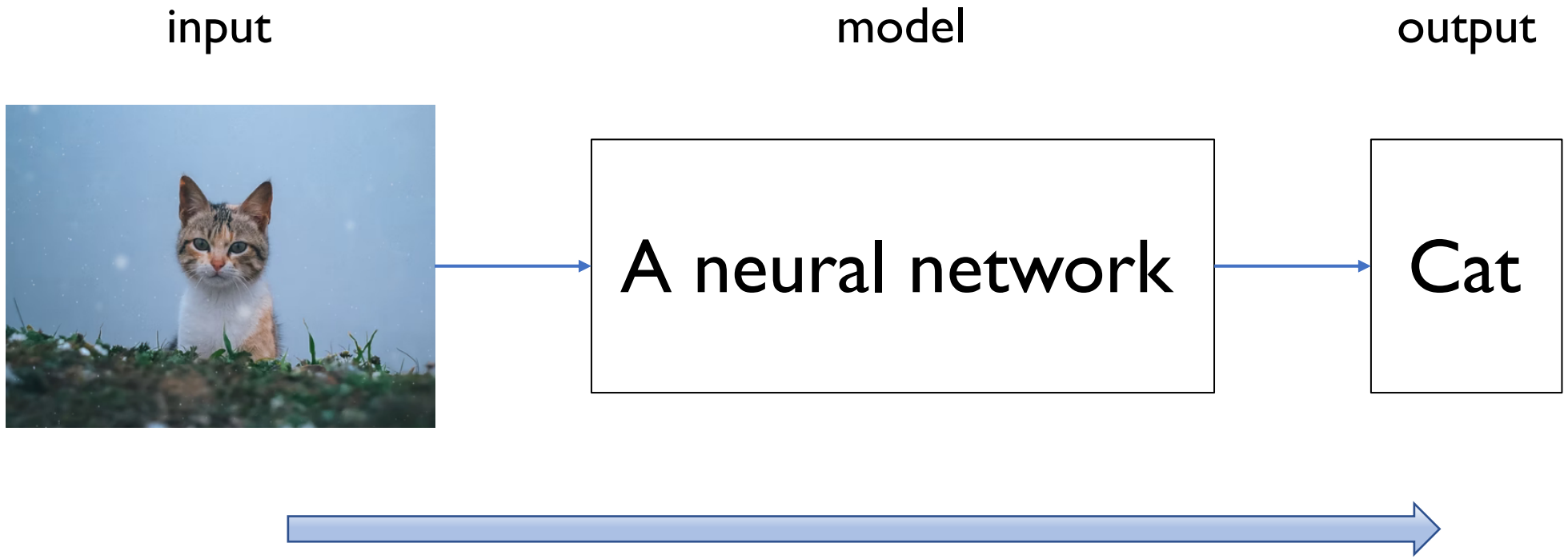
- Background
- Model inversion attacks on images: an overview
- Model inversion attack on graphs: recent advances
- Summary

Outline

- Background
  - Q1: what is model inversion attack?
- Model inversion attacks on images: an overview
- Model inversion attack on graphs: recent advances
- Summary

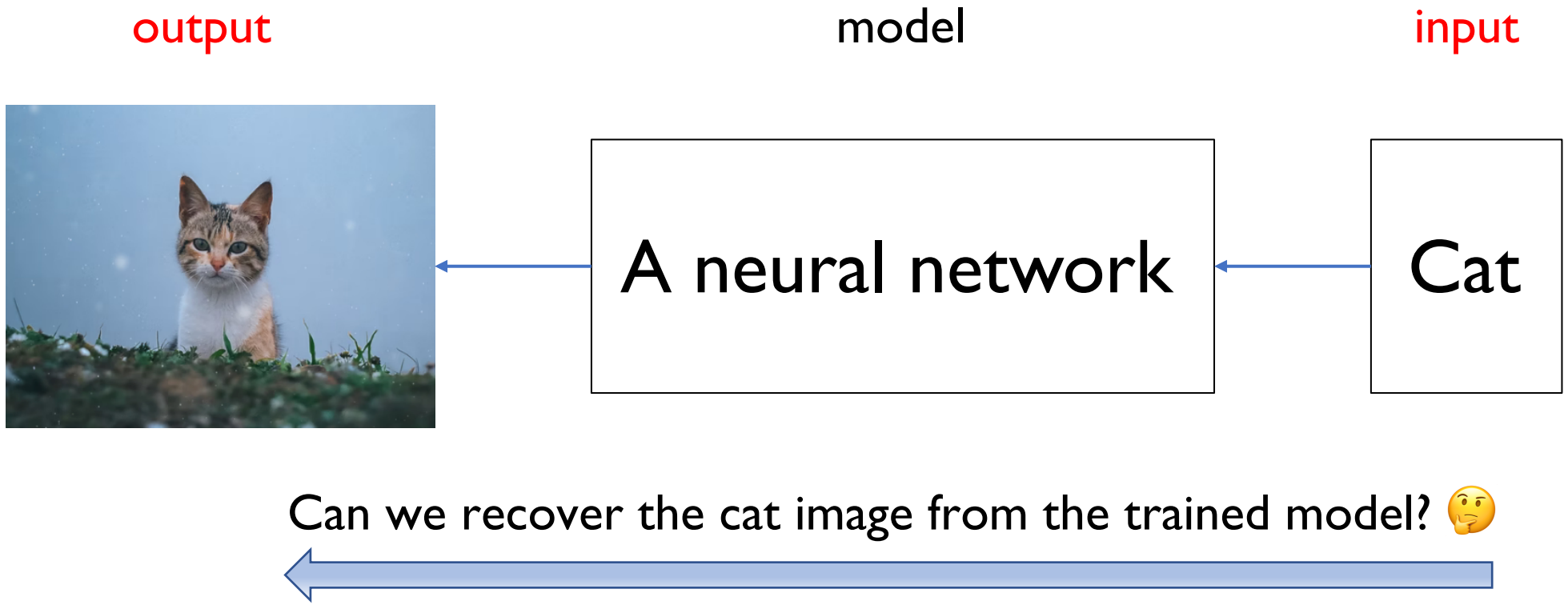
Background | Image classification

Training pipeline of a neural network:



Background | Image classification

What if we reverse the pipeline?



Background | Image classification

What if we reverse the pipeline?

output | model | input

A neural network

Cat

Can we recover the cat image from the trained model? 🤔

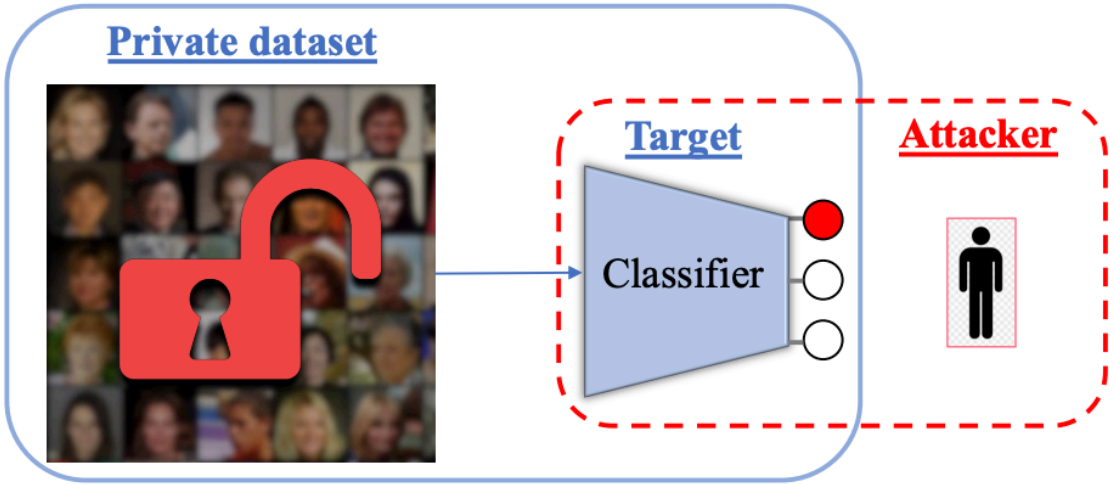
➔ Yes! we can recover the training data via model inversion attack

Background | Research problem

Training: $f_{\boldsymbol{\theta}}(x) \rightarrow y$
Inversion attack: $f_{\boldsymbol{\phi}}^{-1}(y) \rightarrow x$

Definition of model inversion attack (aka MIA, MI attack)

- a malicious user attempts to **recover** the private data that is used to **train** a neural network (i.e., **the target model**)



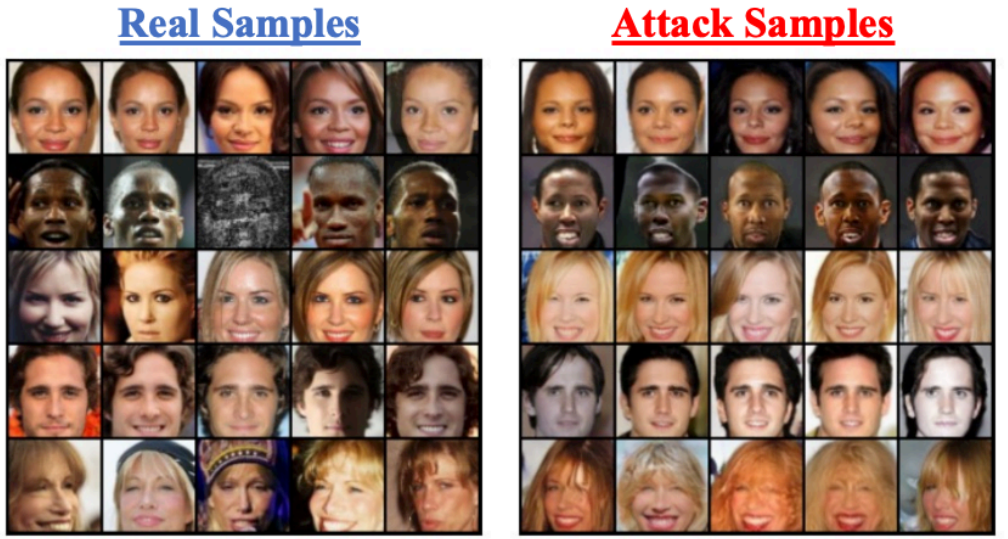


Image from Variational Model Inversion Attacks. NeurIPS 2021.

Outline

- Background
  - Q1: what is model inversion attack?
- An overview of model inversion attacks on images
  - Q2: how to recover the images used for training?
- Model inversion attack on graphs
- Summary

Pioneer works

MI attack with **simple** models

- e.g., linear regression or decision trees

MI attack is **feasible**

Figure 1: An image recovered using a new model inversion attack (left) and a training set image of the victim (right). The attacker is given only the person's name and access to a facial recognition system that returns a class confidence score.

Privacy in Pharmacogenetics: An End-to-End Case Study of Personalized Warfarin Dosing. USENIX Security 2014.
Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures. CCS 2015.

Pioneer works

MI attack with **simple** models

- e.g., linear regression or decision trees

MI attack is **feasible**

However

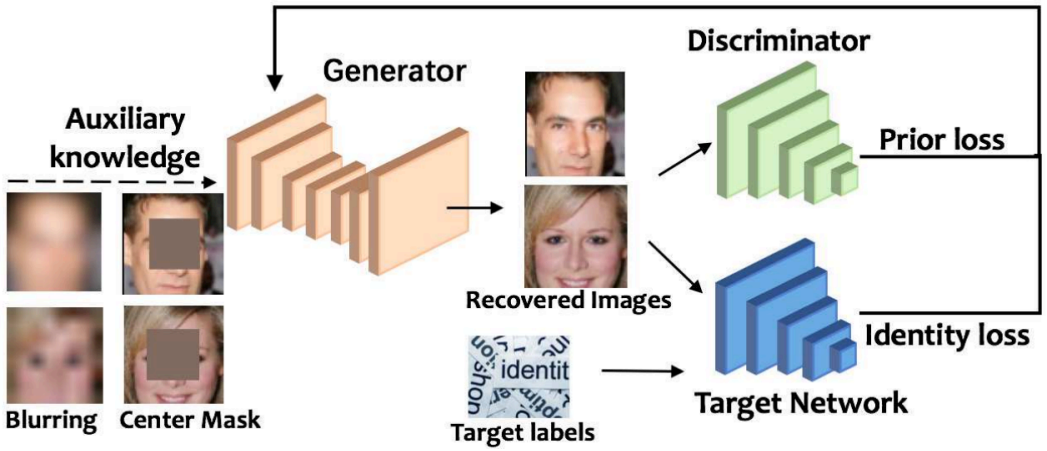
- the reconstructed images are **unclear**
- such a method is **poor** for deep models

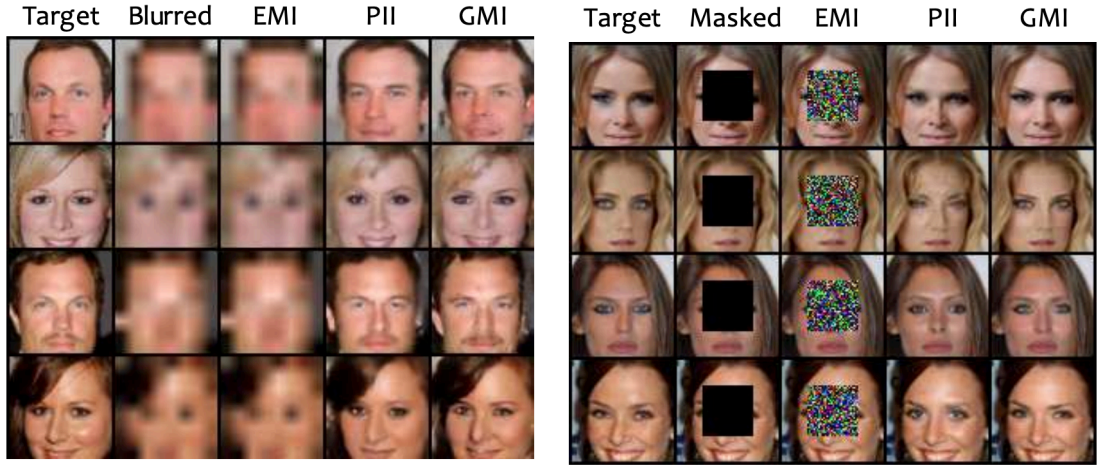
Figure 1: An image recovered using a new model inversion attack (left) and a training set image of the victim (right). The attacker is given only the person's name and access to a facial recognition system that returns a class confidence score.

Privacy in Pharmacogenetics: An End-to-End Case Study of Personalized Warfarin Dosing. USENIX Security 2014.
Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures. CCS 2015.

Generative model inversion

- the **first** to conduct MIA on deep models
  - i.e., the convolution neural networks
- the inversion process is guided by **a distributional prior**
  - through pretrained a generative adversarial network (GAN)
  - for obtaining the **generic knowledge** of human faces.





The Secret Revealer: Generative Model-Inversion Attacks Against Deep Neural Networks. CVPR 2020.

Take a break ☕

Training: $f_{\boldsymbol{\theta}}(x) \rightarrow y$
Inversion attack: $f_{\boldsymbol{\phi}}^{-1}(y) \rightarrow x$

the top-one identification accuracy of face images
inverted from the classifier is only 45%

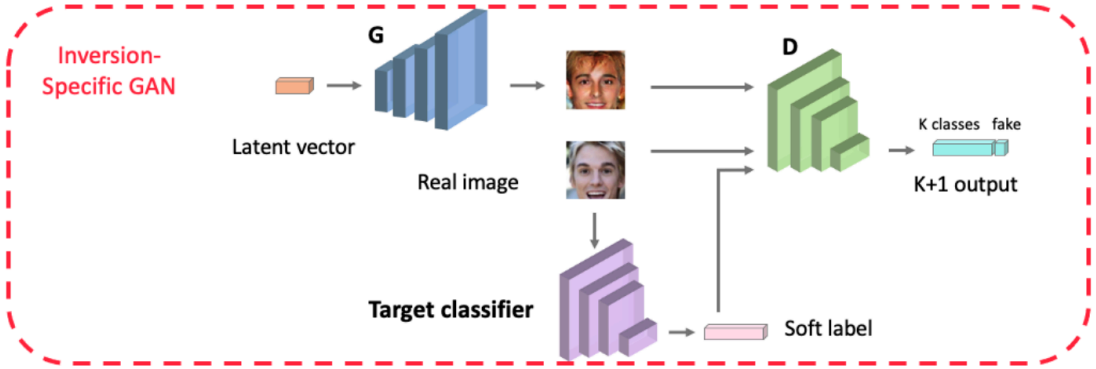
is it because CNNs do not memorize much about private data
or it is due to the imperfect attack algorithm? 🤔

➔ The target network maybe not be fully utilized

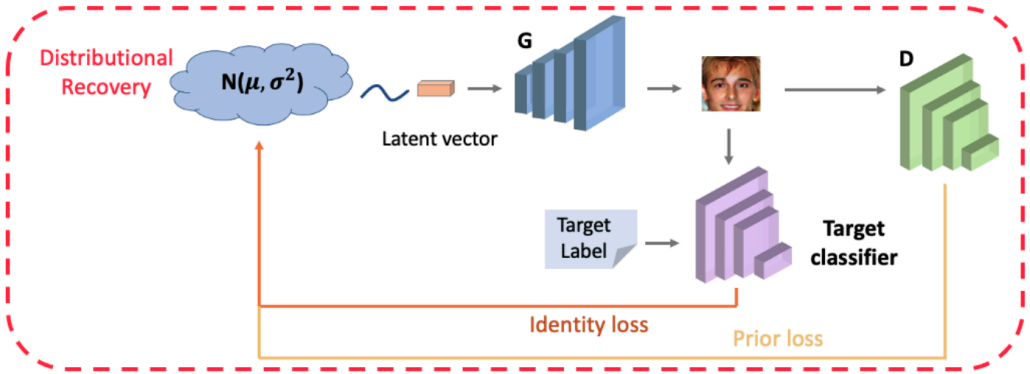
Knowledge-enriched distributional MI

To further distill the useful knowledge from the target model

- [design1] utilizes the target model to generate soft labels
  - for supervising the GAN
- [design2] explicitly parameterizes the distribution of private data
  - proceed MI attacks in a many-to-one way



Step 1. Build an inversion-specific GAN to distill private information



Step 2. Recover the distribution of private domain

Knowledge-Enriched Distributional Model Inversion Attacks. ICCV 2021.

Take a break ☕

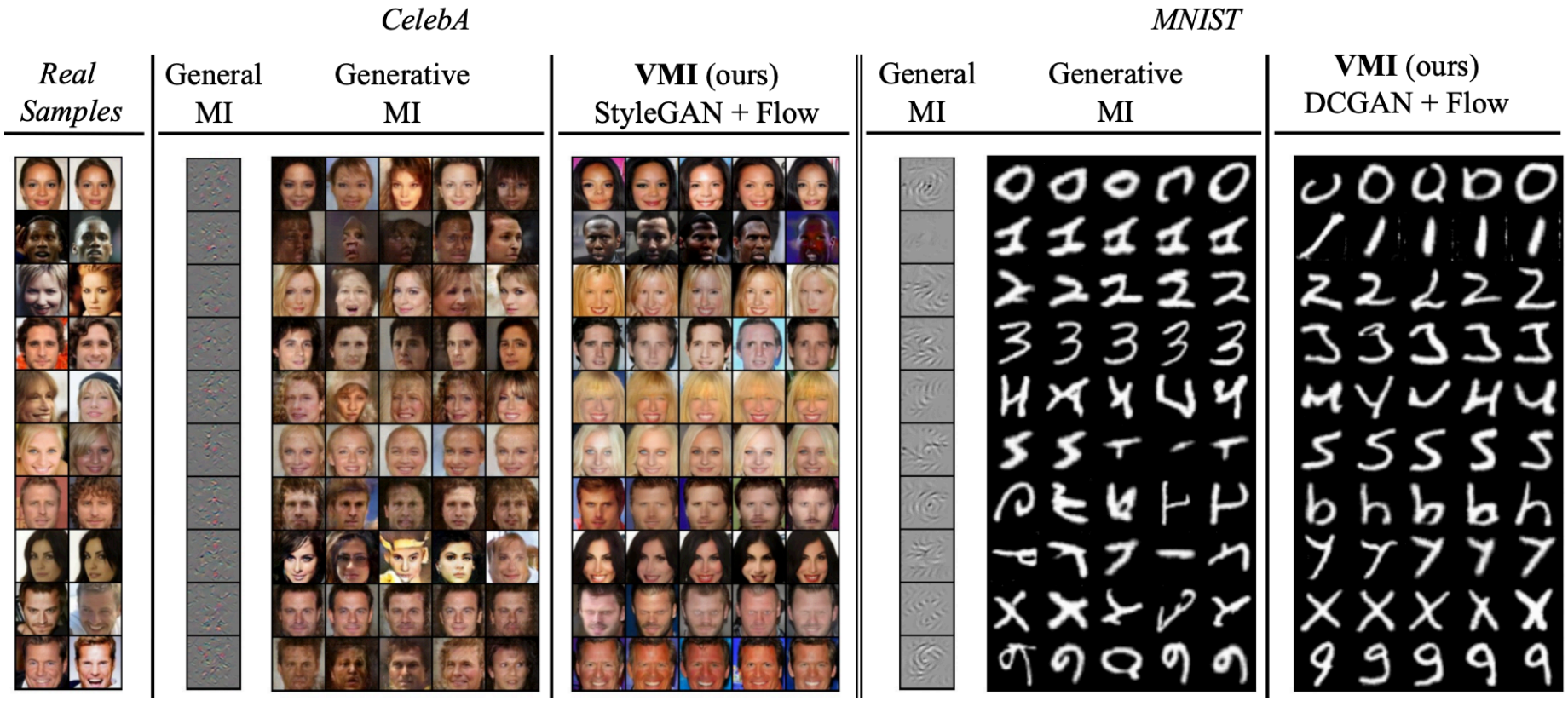
a successful attack should generate **realistic** and **diverse** samples

so, how can we generate **the more diverse** samples? 🤔

Variation model inversion

$$
\begin{aligned}
q^*(\mathbf{x}) &= \arg\min_{q \in \mathcal{Q}_\mathbf{x}} \left\{ D_{\mathrm{KL}}(q(\mathbf{x}) || p_{\mathrm{TAR}}(\mathbf{x}|y)) \right\} \\
&= \arg\min_{q \in \mathcal{Q}_\mathbf{x}} \left\{ \mathbb{E}_{q(\mathbf{x})}[-\log p_{\mathrm{TAR}}(y|\mathbf{x})] + D_{\mathrm{KL}}(q(\mathbf{x}) || p_{\mathrm{TAR}}(\mathbf{x})) + \log p_{\mathrm{TAR}}(y) \right\} \\
&= \arg\min_{q \in \mathcal{Q}_\mathbf{x}} \left\{ \mathbb{E}_{q(\mathbf{x})}[-\log p_{\mathrm{TAR}}(y|\mathbf{x})] + D_{\mathrm{KL}}(q(\mathbf{x}) || p_{\mathrm{TAR}}(\mathbf{x})) \right\}.
\end{aligned}
$$

- formulate a variational objective for both diversity and accuracy



Variational Model Inversion Attacks. NeurIPS 2021.

A short summary

1. A well-defined research problem
2. Effective objectives and mechanisms
3. State-of-the-arts solutions

model inversion attack

- utilize prior knowledge (public data)
- extract knowledge from target model
- generate more realistic samples
- generate more diverse samples

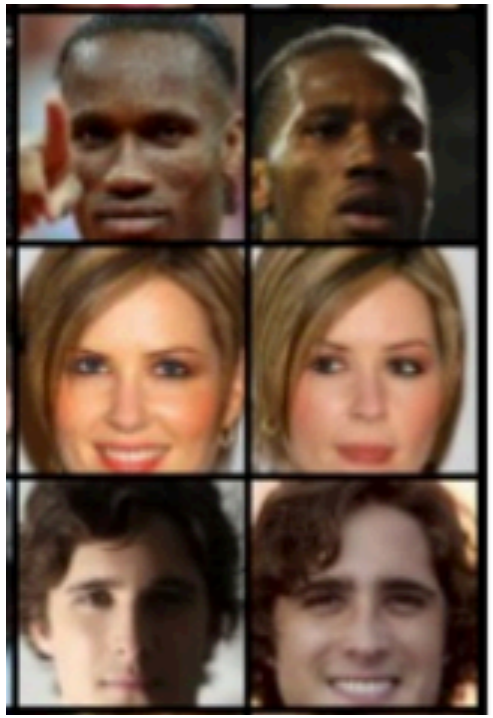
- generative model inversion
- knowledge-enriched distributional MI
- variation model inversion

Outline

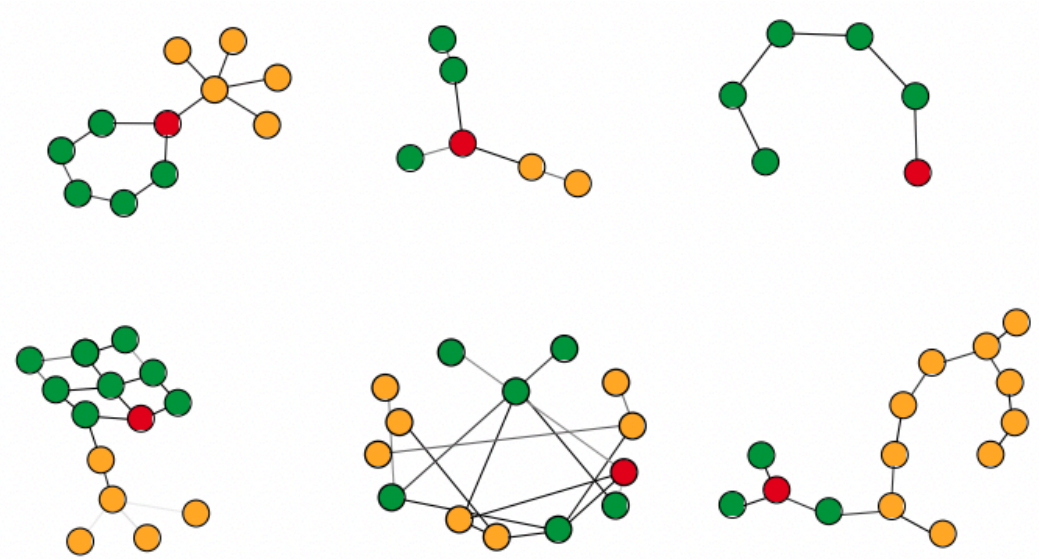
- Background
  - Q1: what is model inversion attack?
- Model inversion attacks on images: an overview
  - Q2: how to recover the images used for training?
- Model inversion attack on graphs: recent advances
  - Q3: what can be attacked for graph data?
  - Q4: how to conduct such an attack on graph data?
- Summary

MI attack | from images to graphs

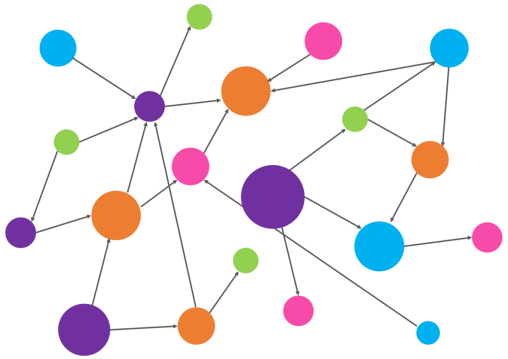
"human faces"

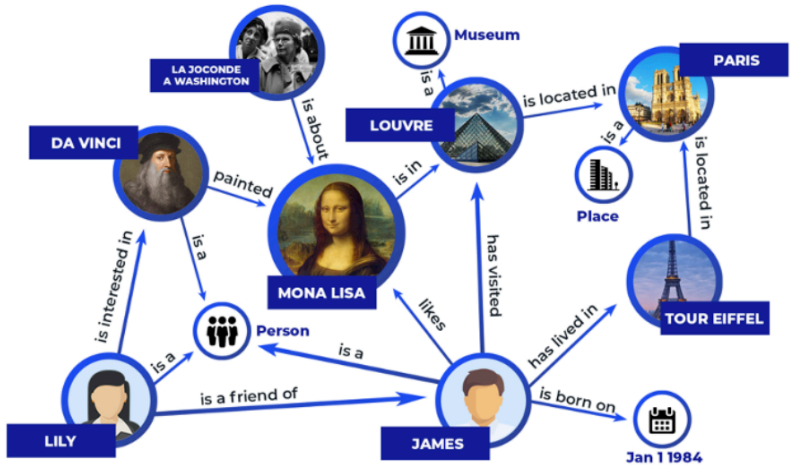


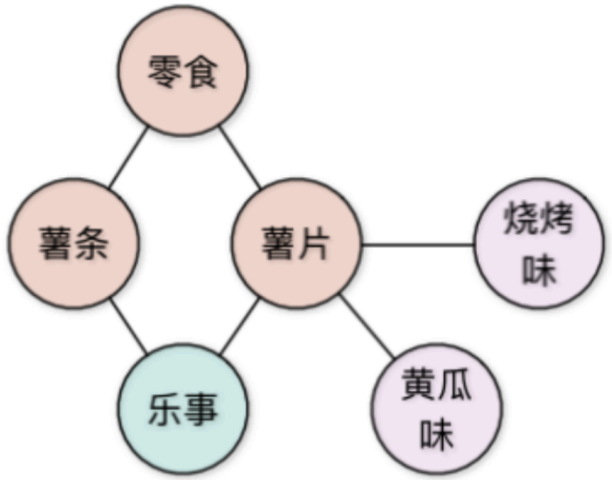
but, what about "graphs"?



MI attack | from images to graphs

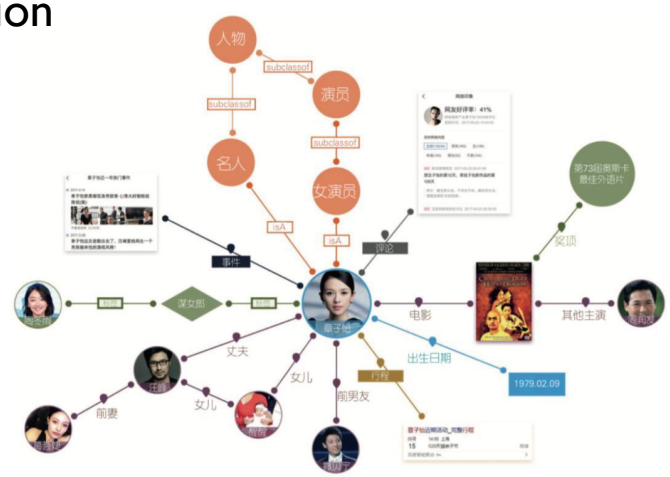


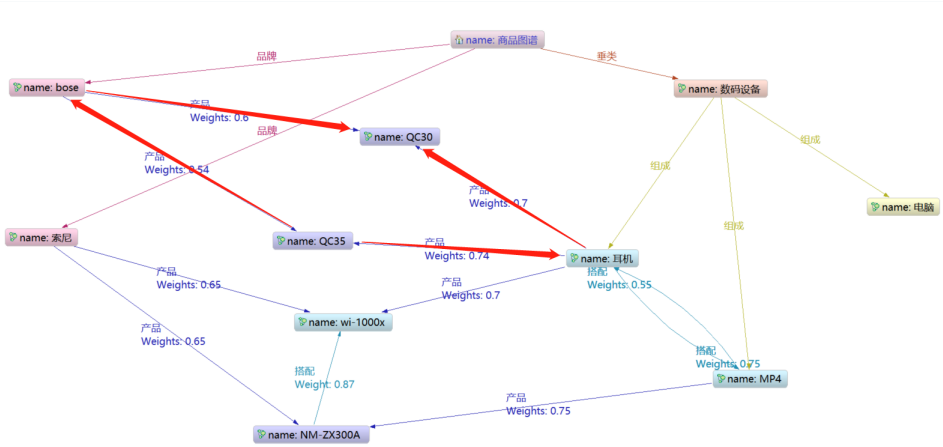




Graph is a ***general*** form of data expression

- ***diverse*** and ***domain-specific***
  - molecules
  - social networks
  - citation networks

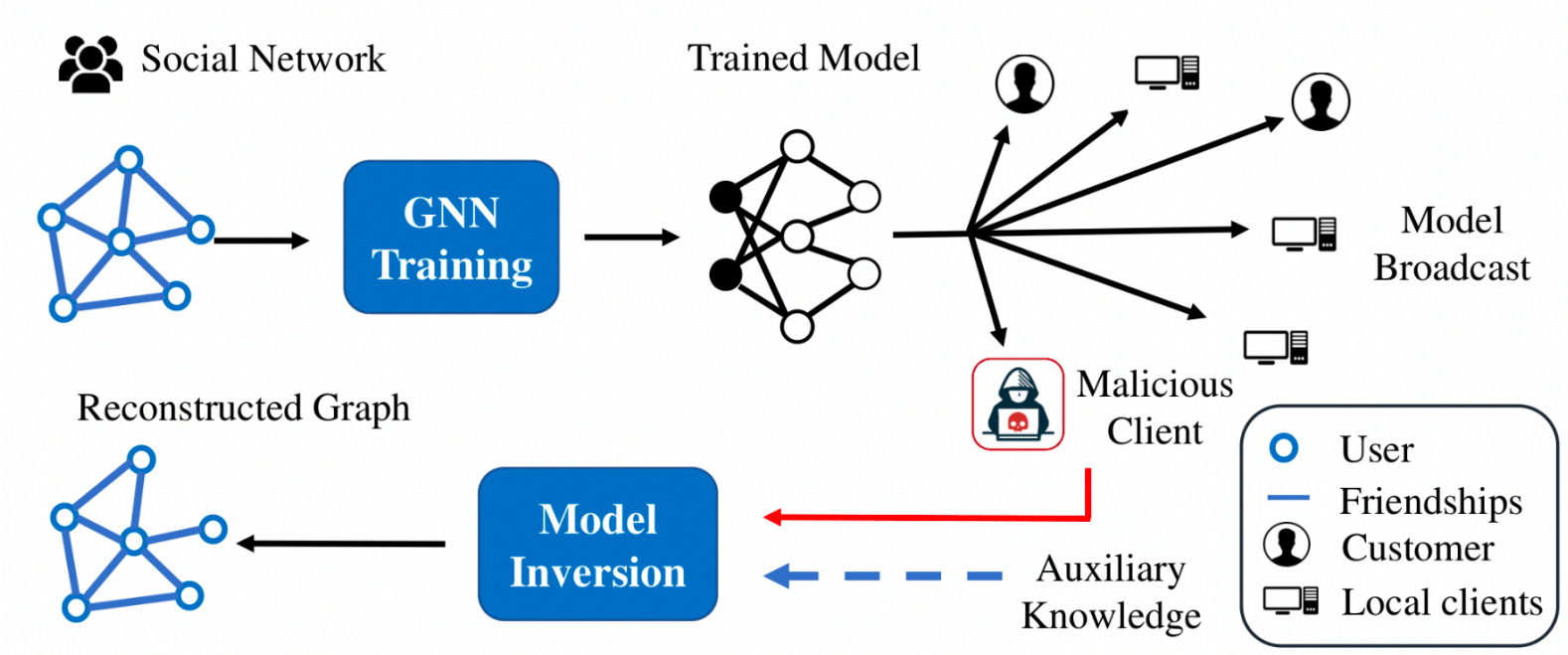




MI attack on graphs | motivation

Why Graph Reconstruction Attack?

- inferring links leads to a severe ***privacy threat*** when the links represent ***sensitive*** information
  - e.g., the relationship between users in social networks
- it may also leak the model owner's ***intellectual property***



GraphMI: Extracting Private Graph Data from Graph Neural Networks. IJCAI 2021.

MI attack on graphs | challenges

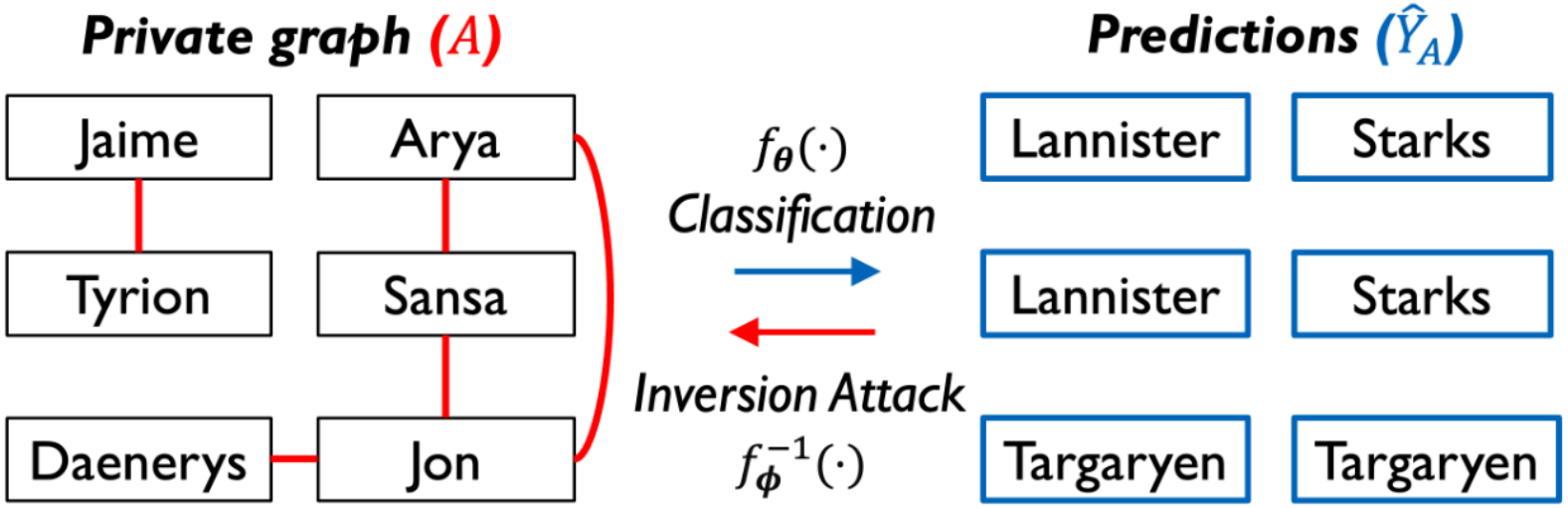
Key challenges

- Lack of domain knowledge as the **priors**
  - graphs are less intuitive than images
  - the domain knowledge can be diverse

- The **discrete** nature of graph structure
  - hard to optimize in a differentiable way
  - the nodes and edges cannot be resized to the same shape

MI attack on graphs | an example

Graph Reconstruction Attack (GRA): a kind of MI attack on graphs



the forward inference is to predict the node category $\hat{Y}_A$

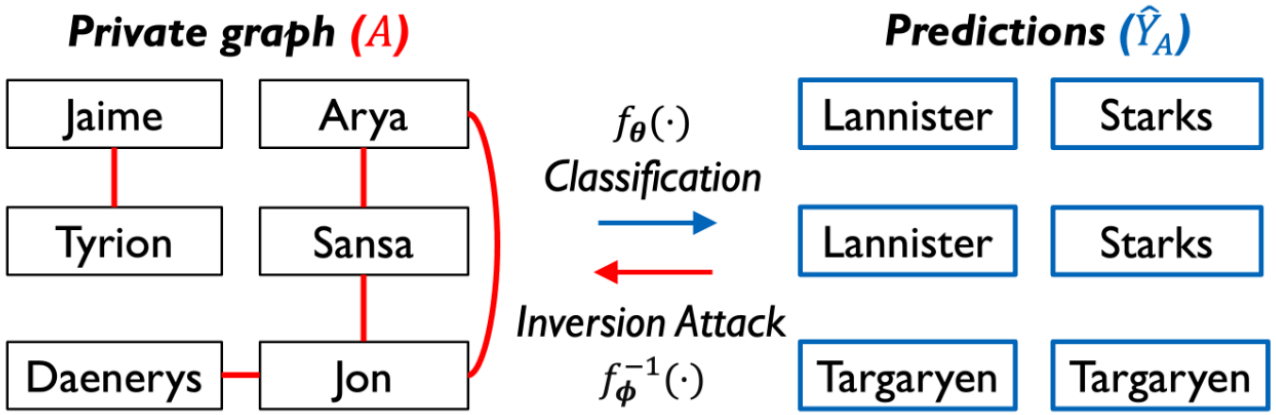
- i.e., ***family*** of each person

the backward attack is to recover the adjacency A

- i.e., ***kinship*** among the persons

MI attack on graphs | definition

Graph Reconstruction Attack (GRA):
given a set of prior knowledge $\mathcal{K}$ and a trained GNN $f_{\boldsymbol{\theta}}(\cdot)$
the GRA aims to recover the adjacency $\hat{A}$ of the training graph (A, X)



training with $f_{\boldsymbol{\theta}}$: $(A, X) \rightarrow \cdots \rightarrow \boldsymbol{H} \rightarrow \widehat{\boldsymbol{Y}} \leftrightarrow Y$
reconstruction attack with $f_{\boldsymbol{\phi}}^{-1}$: $\mathcal{K} \rightarrow \hat{A} \leftrightarrow A$

$$\hat{A}^* = \text{argmax}_{\hat{A}} \mathbb{P}(\hat{A} | \mathcal{K})$$

- $\hat{A}^*$ is the recovered adjacency
- $\mathcal{K}$: set of prior knowledge
  - $\mathcal{K} \subseteq \{f_{\boldsymbol{\theta}}(\cdot), X, \boldsymbol{H}, \widehat{\boldsymbol{Y}}, Y, D_{aux}\}$
  - where D_{aux} is an auxiliary data
- $\mathbb{P}(\cdot)$: attack method to generate $\hat{A}$

Outline

- Background
  - Q1: what is model inversion attack?
- Model inversion attacks on images: an overview
  - Q2: how to recover the images used for training?
- Model inversion attack on graphs: recent advances
  - Q3: what can be attacked for graph data?
  - Q4: how to conduct such an attack on graph data?
- Summary

MI attack on graphs | existing works

training with $f_{\boldsymbol{\theta}}$: $(A, X) \rightarrow \cdots \rightarrow \boldsymbol{H} \rightarrow \widehat{\boldsymbol{Y}} \leftrightarrow Y$

reconstruction attack with $f_{\boldsymbol{\phi}}^{-1}$: $\mathcal{K} \rightarrow \hat{A} \leftrightarrow A$

A taxonomy

- Class1: non-learnable attacks with single dataset
  - $\mathcal{K} = \{\boldsymbol{H}, \widehat{\boldsymbol{Y}}\}$ (model's outputs)
- Class2: learnable attacks with single dataset
  - $\mathcal{K} = \{f_{\boldsymbol{\theta}}(\cdot), X, Y\}$ (model + data)
- Class3: learnable attacks with dual datasets
  - $\mathcal{K} = \{\boldsymbol{H}, D_{aux}\}$ (model's outputs + auxiliary data)

$$\hat{A}^* = \text{argmax}_{\hat{A}} \mathbb{P}\left(\hat{A} | \mathcal{K}\right)$$

- $\hat{A}^*$ is the recovered adjacency
- $\mathcal{K}$: set of prior knowledge
  - $\mathcal{K} \subseteq \{f_{\boldsymbol{\theta}}(\cdot), X, \boldsymbol{H}, \widehat{\boldsymbol{Y}}, Y, D_{aux}\}$
  - where D_{aux} is an auxiliary data
- $\mathbb{P}(\cdot)$: attack method to generate $\hat{A}$

Basically, Class1 < Class2 < Class3 (in attack effectiveness)

MI attack on graphs | Class1: non-learnable attacks

***Non-learnable* attacks with *single* dataset**

- $\mathcal{K} = \{\boldsymbol{H}, \widehat{\boldsymbol{Y}}\}$ (model's outputs)
- key: similarity between nodes
- $\hat{A} = distance(\boldsymbol{H})$
- where $\hat{A}_{ij} = distance(\boldsymbol{h}_i, \boldsymbol{h}_j)$

Metrics	Definition
Cosine	$1 - \frac{f(u) \cdot f(v)}{\|f(u)\|_2 \|f(v)\|_2}$
Euclidean	$\|f(u) - f(v)\|_2$
Correlation	$1 - \frac{(f(u) - \overline{f(u)}) \cdot (f(v) - \overline{f(v)})}{\|(f(u) - \overline{f(u)})\|_2 \|(f(v) - \overline{f(v)})\|_2}$
Chebyshev	$\max_i \lvert f_i(u) - f_i(v) \rvert$
Braycurtis	$\frac{\sum \lvert f_i(u) - f_i(v) \rvert}{\sum \lvert f_i(u) + f_i(v) \rvert}$
Manhattan	$\sum_i \lvert f_i(u) - f_i(v) \rvert$
Canberra	$\sum_i \frac{\lvert f_i(u) - f_i(v) \rvert}{\lvert f_i(u) \rvert + \lvert f_i(v) \rvert}$
Sqeuclidean	$\|f(u) - f(v)\|_2^2$

Stealing Links from Graph Neural Networks. USENIX Security 2020.

MI attack on graphs | Class2: learnable attacks with single dataset

***Learnable* attacks with *single* dataset**

- $\mathcal{K} = \{f_{\boldsymbol{\theta}}(\cdot), X, Y\}$
- key: minimize the classification error
- $\hat{A}^* = \text{argmin}_{\hat{A}} \mathcal{L}_{cls}(f_{\boldsymbol{\theta}}(X, \hat{A}), Y)$

GraphMI: Extracting Private Graph Data from Graph Neural Networks. IJCAI 2021.

MI attack on graphs | Class2: learnable attacks with single dataset

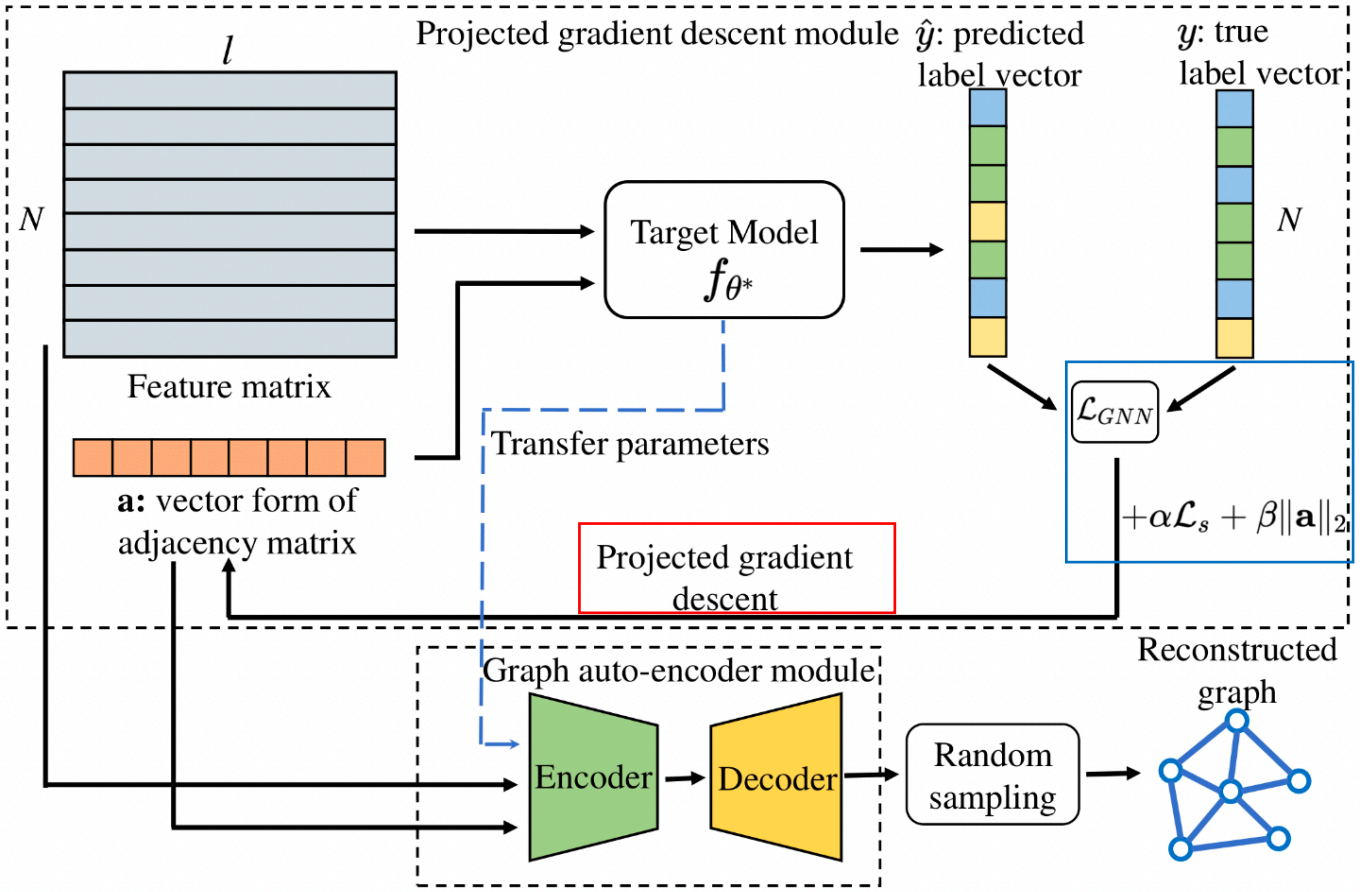


Figure 2: Overview of GraphMI

Objective:
minimize the loss between true y and predicted y

$$\min_{A\in\{0,1\}^{N\times N}} \mathcal{L}_{GNN}(A) = \frac{1}{N}\sum_{i=1}^{N} \ell_i(A, f_{\theta^*}, \mathbf{x}_i, y_i)$$

$$s.t. \quad A = A^\top.$$

added with feature smoothness (homophily) and regularization on A

The continuous optimization problem 11 is solved by projected gradient descent (PGD):

$$\mathbf{a}^{t+1} = P_{[0,1]}\left[\mathbf{a}^t - \eta_t g_t\right], \tag{12}$$

where t is the iteration index of PGD, η_t is the learning rate, g_t is the gradients of loss $\mathcal{L}_{attack}$ in 10 evaluated at $\mathbf{a}^t$, and

$$P_{[0,1]}[x] = \begin{cases} 0 & x < 0 \\ 1 & x > 1 \\ x & otherwise \end{cases} \tag{13}$$

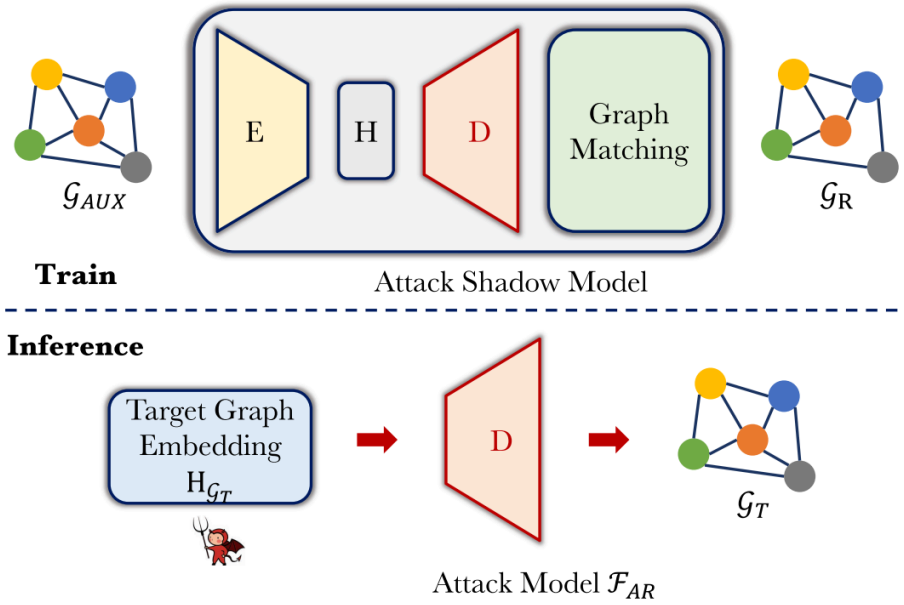
is the projection operator.

GraphMI: Extracting Private Graph Data from Graph Neural Networks. IJCAI 2021.

MI attack on graphs | Class3: transferable attacks with dual datasets

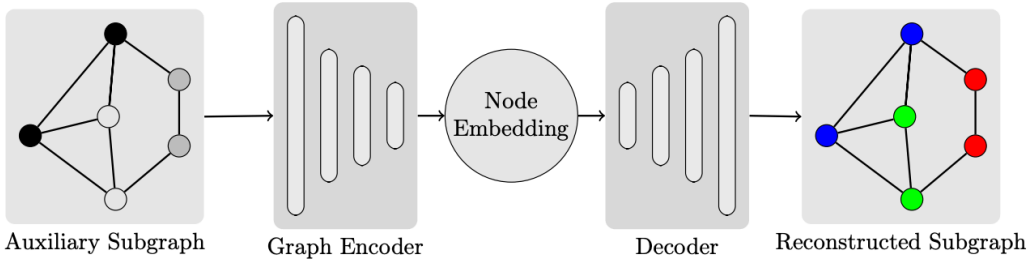
***Learnable* attacks with *dual* datasets**

- $\mathcal{K} = \{\boldsymbol{H}, D_{aux}\}$
- key: use D_{aux} to train a encoder-decoder $f_{\boldsymbol{\phi}}$
  - $D_{aux} = (A_{aux}, X_{aux})$
  - $f_{\boldsymbol{\phi}}: (A_{aux}, X_{aux}) \rightarrow \boldsymbol{H}_{aux} \rightarrow A_{aux}$
- then, directly apply $f_{\boldsymbol{\phi}}$ on $\boldsymbol{H}$
  - $\hat{A} = f_{\boldsymbol{\phi}}^{dec}(\boldsymbol{H})$

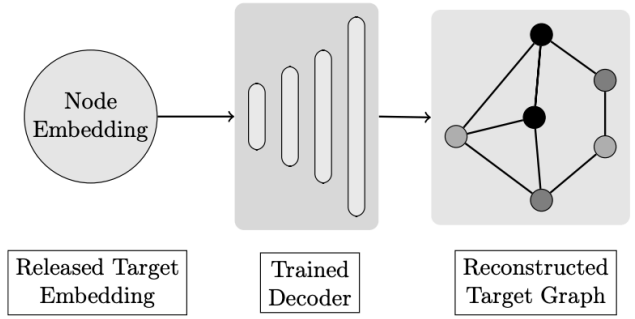


MI attack on graphs | Class3: transferable attacks with dual datasets

[1]

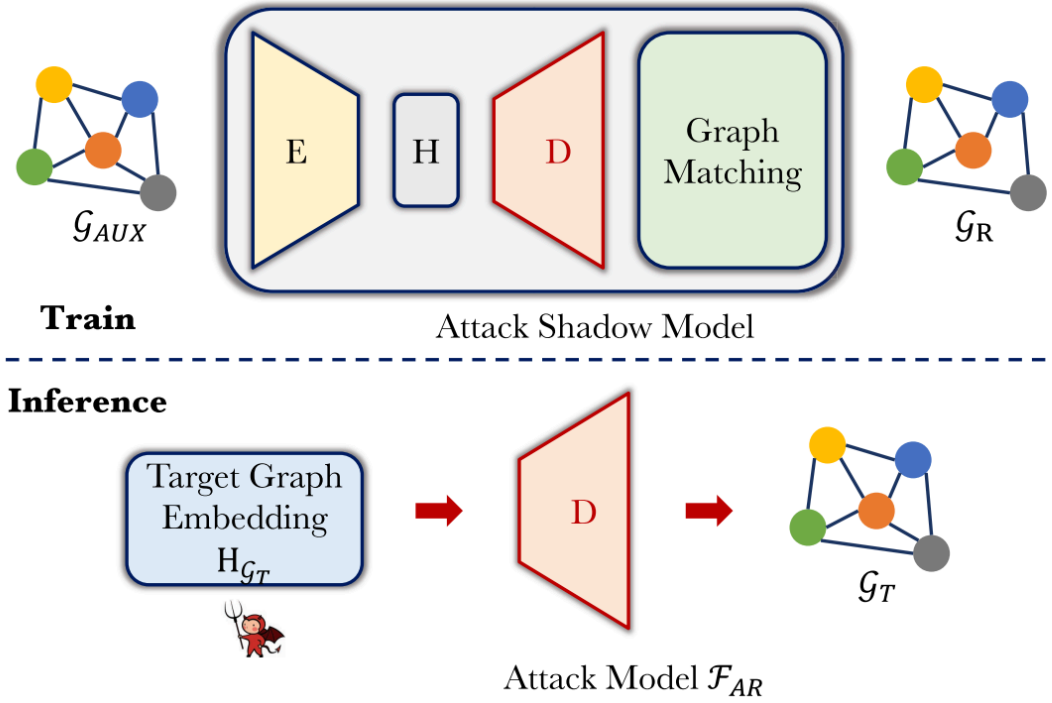


(a) Adversary trains attack model on auxiliary subgraph



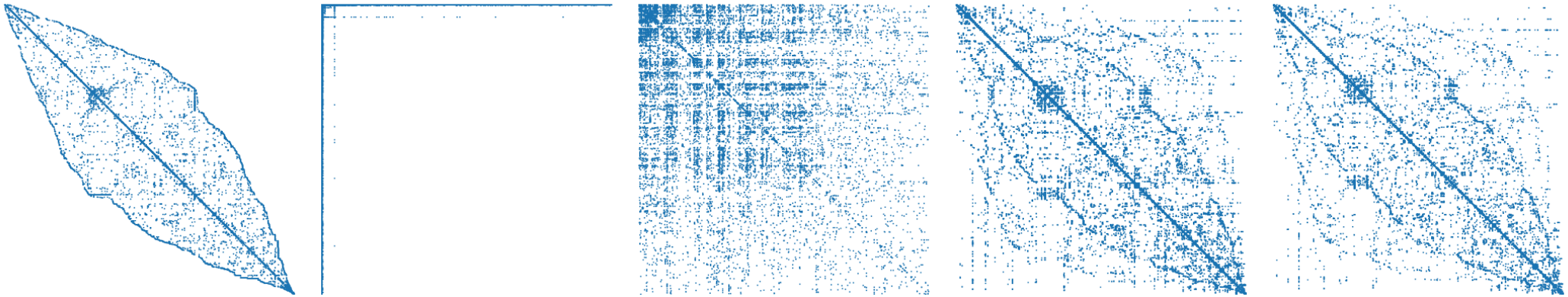
(b) Attack model reconstructs target graph

[2]



[1] Quantifying Privacy Leakage in Graph Embedding. MobiQuitous 2020.
[2] Inference Attacks Against Graph Neural Networks. USENIX Security 2022.

MI attack on graphs | Effectiveness



(a) original (b) direct recovery (c) invert embedding (d) *k*NN graph (e) MNEMON

Figure 7: Bitmap visualization of the recovered graphs by all baselines and MNEMON. MNEMON, visually, removes fair amount of false positive edges.

Finding MNEMON: Reviving Memories of Node Embeddings. CCS 2022.

MI attack on graphs | Effectiveness

Table 7: Average AUC with standard deviation for Attack-4 on all the 8 datasets. Best results are highlighted in bold.

Target Dataset	Shadow Dataset: AIDS	COX2	DHFR	ENZYMES	PROTEINS_full	Citeseer	Cora	Pubmed
AIDS	-	0.750 ± 0.009	**0.763 ± 0.010**	0.733 ± 0.007	0.557 ± 0.009	0.729 ± 0.015	0.702 ± 0.010	0.673 ± 0.009
COX2	0.802 ± 0.031	-	**0.866 ± 0.004**	0.782 ± 0.012	0.561 ± 0.030	0.860 ± 0.002	0.853 ± 0.004	0.767 ± 0.023
DHFR	0.758 ± 0.022	**0.812 ± 0.005**	-	0.662 ± 0.030	0.578 ± 0.067	0.799 ± 0.002	0.798 ± 0.009	0.736 ± 0.005
ENZYMES	**0.741 ± 0.010**	0.684 ± 0.024	0.670 ± 0.008	-	0.733 ± 0.019	0.624 ± 0.002	0.627 ± 0.014	0.691 ± 0.012
PROTEINS_full	0.715 ± 0.009	0.802 ± 0.025	0.725 ± 0.041	0.863 ± 0.010	-	0.784 ± 0.031	0.815 ± 0.012	**0.867 ± 0.003**
Citeseer	0.832 ± 0.078	0.940 ± 0.005	0.914 ± 0.007	0.879 ± 0.062	0.833 ± 0.088	-	**0.967 ± 0.001**	0.955 ± 0.003
Cora	0.572 ± 0.188	0.899 ± 0.003	0.887 ± 0.014	0.878 ± 0.045	0.738 ± 0.168	**0.945 ± 0.001**	-	0.924 ± 0.005
Pubmed	0.777 ± 0.056	0.893 ± 0.001	0.90 ± 0.006	0.866 ± 0.002	0.806 ± 0.042	**0.907 ± 0.004**	0.902 ± 0.001	-

Stealing Links from Graph Neural Networks. USENIX Security 2020.

Outline

- Background
  - Q1: what is model inversion attack?
- Model inversion attacks on images: an overview
  - Q2: how to recover the images used for training?
- Model inversion attack on graphs: recent advances
  - Q3: what can be attacked for graph data?
  - Q4: how to conduct such an attack on graph data?
- Summary

Take home messages

Model inversion attack from images to graphs

- Q1: what is model inversion attack
  ➔ **recover** the private dataset from the target model

- Q2: how to recover the images used for training
  ➔ **utilize** prior knowledge from public data and **extract** knowledge from target model

- Q3: what can be attacked for graph data?
  ➔ to reconstruct the **adjacency** of the training graph

- Q4: how to conduct reconstruction attack on graph data?
  ➔ utilize the **prior knowledge** and conduct **learnable/non-learnable** attack

Future directions

Directions

- utilize more prior knowledge (if available)
- inspecting the recovered graph's properties
- robust methods for defending against the attack

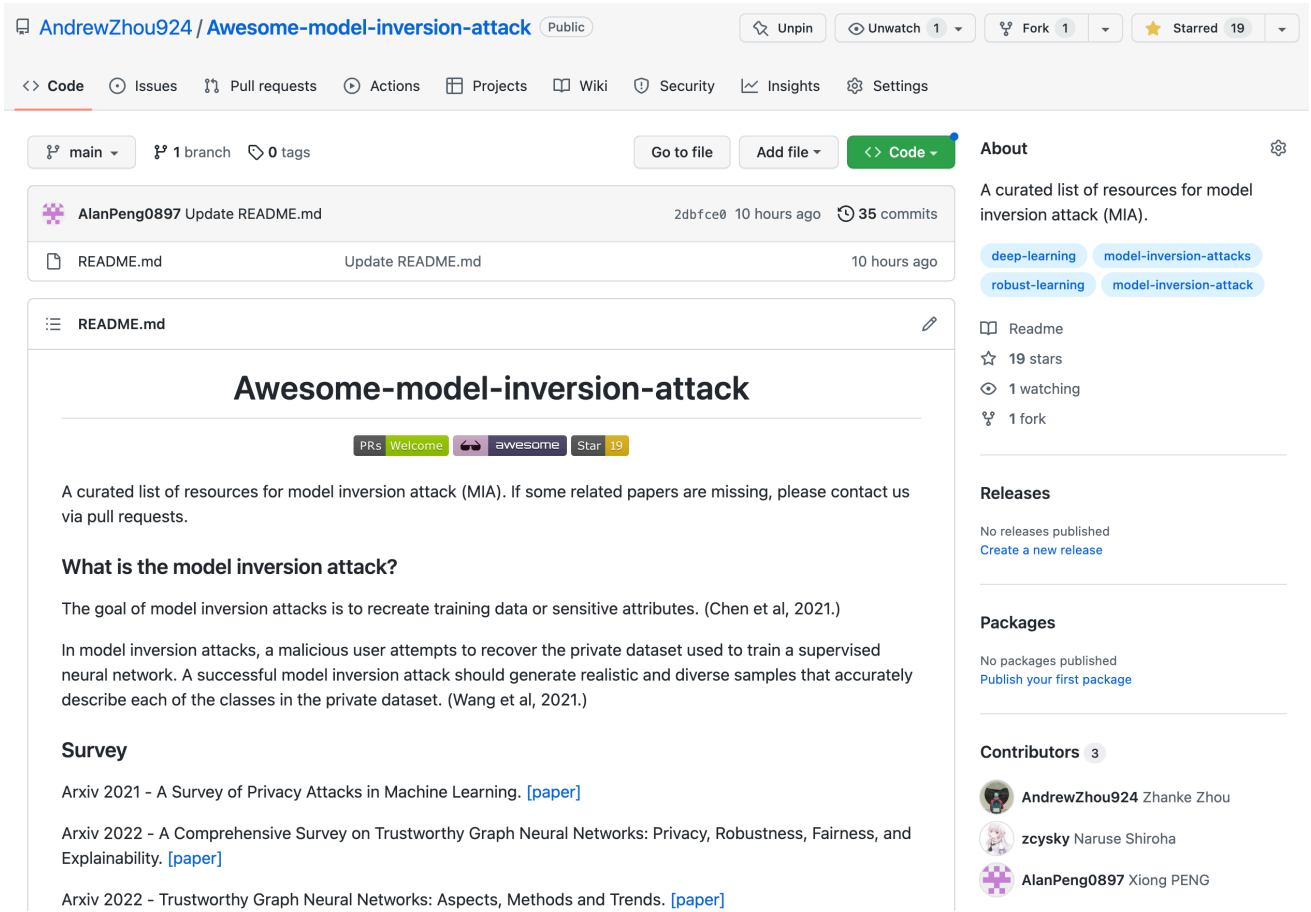
General principles

- to attack, you must extract more
- to defense, you must forget more

A curated list of resources

include 60+ paper of 3 domains

- computer vision
- natural language processing
- graph learning



https://github.com/AndrewZhou924/Awesome-model-inversion-attack

Q&A

Thanks for your attention!

Potential risk and values

- The MI attack approaches can be **misused** to attack real-world targets

- However, it is important to **raise the awareness** of such an attack
  - inform the community about the risk of privacy leaks, especially the user side
  - e.g., the attack manners and patterns

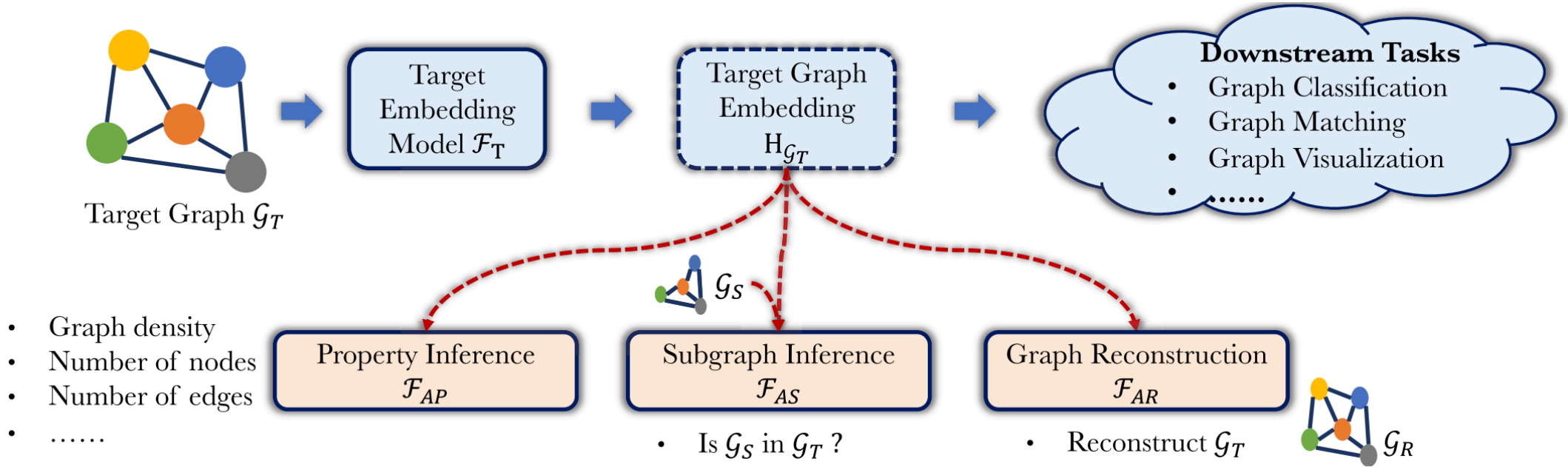
- More importantly, use the MI attacks to inspire the **robust method**
  - to develop the defensing strategies and to better protect privacy
  - to make the AI products more safe and trustworthy

Potential risk and values

"The gun is not guilty, the person who pulled the trigger is."

----- by Mikhail Kalashnikov, farther of AK-47

Appendix: evaluation metrics



Inference Attacks Against Graph Neural Networks. USENIX Security 2022.

Appendix: privacy attack on graph

3.1.1 Types of Privacy Attacks on GNNs. The goal of privacy attacks on GNNs is to extract information that is not intended to be shared. The target information can be about the training graph such as the membership, sensitive attributes of nodes, and connections of nodes. In addition, some attackers aim to extract the model parameters of GNNs. Based on the target knowledge, the privacy attacks can generally be split into four categories:

- **Membership Inference Attack**: In membership inference attack, the attackers try to determine whether a target sample is part of the training set. For example, suppose researchers train a GNN model on social network of COVID-19 patients to analyze the propagation of virus. The membership inference attack can identify if a target subject is in training patient network, resulting in information leakage of the subject. Different from i.i.d data, the format of the target samples can be nodes or graphs. For instance, for node classification task, the target samples can be subgraph of the target node's local graph [137] or only contain the node attributes [75]. For graph classification task, the target sample is a graph to be classified [199].
- **Reconstruction Attack**: Reconstruction attack, also known as model inversion attack, aims to infer the private information of the input graph. Since the graph-structure data is composed of graph topology and node attributes, the reconstruction attack on GNNs can be split into *structure reconstruction*, i.e., infer the structures of target samples, and *attribute reconstruction* (also known as attribute inference attack), i.e., infer the attributes of target samples. Generally, the embeddings of the target samples are required to conduct the reconstruction attack.
- **Property Inference Attack**: Different from attribute reconstruction attack, property inference attack aims to infer dataset properties that are not encoded as features. For instance, one may want to infer the ratio of women and men in a social network, where this information is not contained in node attributes. The attacker may also be interested in structure-related properties such as degrees of a node, which is the number of friends of the target user in a social network [232].
- **Model Extraction Attack**: This attack aims to extract the target model information by learning a model that behaves similarly to the target model. It may focus on different aspects of the model information, which results in two goals in model extraction: (i) The attacker aims to obtain a model that matches the accuracy of the target model; (ii) The attacker tries to replicate the decision boundary of the target model. Model Extraction Attack can threaten the security of model for API service [135] and can be a stepping stone for various privacy attacks and adversarial attacks.

https://arxiv.org/pdf/2204.08570.pdf

Appendix: evaluation metrics

		VMI (ours)									
		DCGAN				StyleGAN					
	Generative MI	$q(\mathbf{z})$ = Gaussian	Flow			Gaussian			Flow		
	[Zhang et al., 2020]	γ=0	0	1e-3	1e-1	0	1e-3	1e-1	0	1e-3	1e-1
Accuracy	0.07± 0.02	0.24± 0.05	0.33± 0.09	0.37± 0.07	0.13± 0.03	0.57± 0.06	0.56± 0.05	0.23± 0.03	**0.58**± 0.06	0.55± 0.06	0.39± 0.07
Precision	0.51± 0.04	0.64± 0.05	0.48± 0.08	0.52± 0.06	0.40± 0.06	0.87± 0.02	0.88± 0.02	0.82± 0.02	0.87± 0.03	0.87± 0.03	**0.89**± 0.03
Density	0.41± 0.04	0.67± 0.08	0.49± 0.11	0.52± 0.08	0.38± 0.06	1.26± 0.07	1.28± 0.07	1.14± 0.06	1.22± 0.08	1.22± 0.08	**1.31**± 0.10
Recall	0.21± 0.04	0.03± 0.01	0.00± 0.00	0.01± 0.01	0.13± 0.03	0.22± 0.03	0.25± 0.03	**0.42**± 0.03	0.11± 0.02	0.15± 0.03	0.21± 0.04
Coverage	0.83± 0.03	0.79± 0.04	0.37± 0.06	0.67± 0.06	0.70± 0.06	0.98± 0.01	0.98± 0.01	**0.98**± 0.01	0.96± 0.02	0.97± 0.02	0.98± 0.02
Diversity	0.52± 0.05	0.41± 0.04	0.19± 0.06	0.34± 0.06	0.41± 0.07	0.60± 0.03	0.61± 0.03	**0.70**± 0.03	0.54± 0.02	0.56± 0.04	0.59± 0.05
FID	43.21	28.98	58.39	40.89	40.74	16.69	16.11	**13.49**	17.28	17.41	21.35

Target Accuracy

- how many generated images can be classified as the target class

Sample Realism

- how real are the generated images

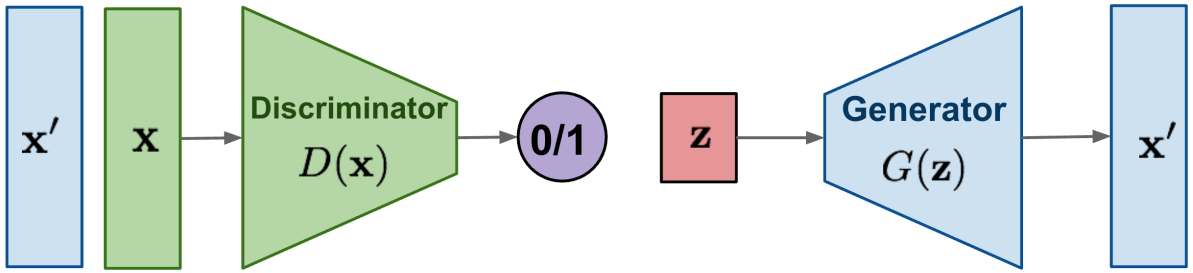
Sample Diversity

- how many images of the same class are recovered

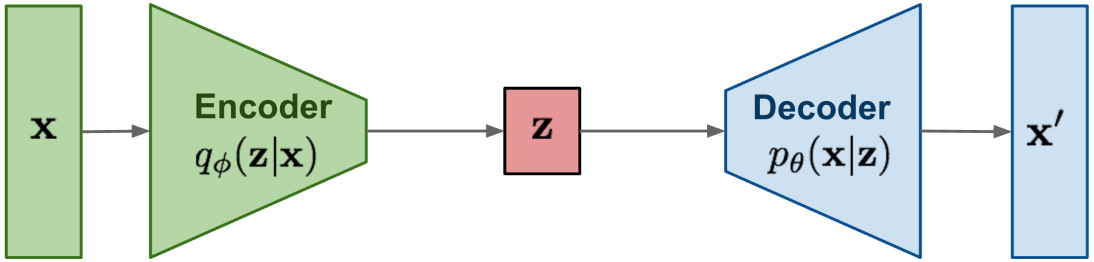
Variational Model Inversion Attacks. NeurIPS 2021.

Appendix: generative models

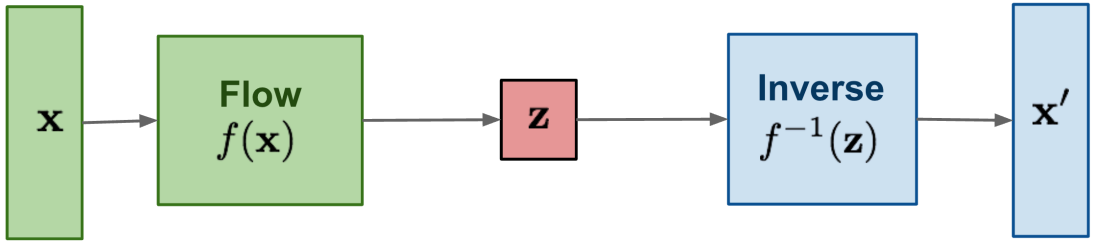
GAN: Adversarial training



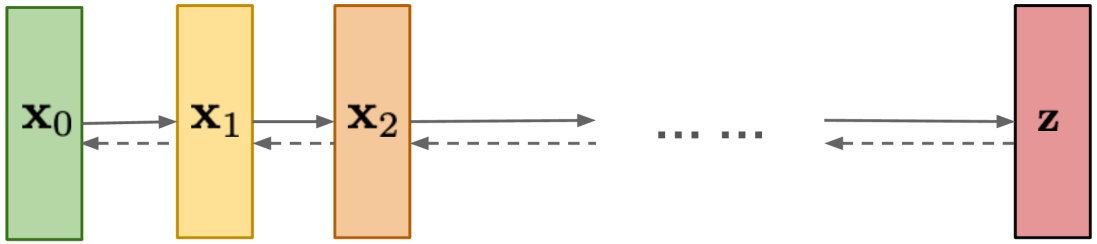
VAE: maximize variational lower bound



Flow-based models: Invertible transform of distributions



Diffusion models: Gradually add Gaussian noise and then reverse



https://lilianweng.github.io/posts/2021-07-11-diffusion-models/

Appendix: diffusion probabilistic model

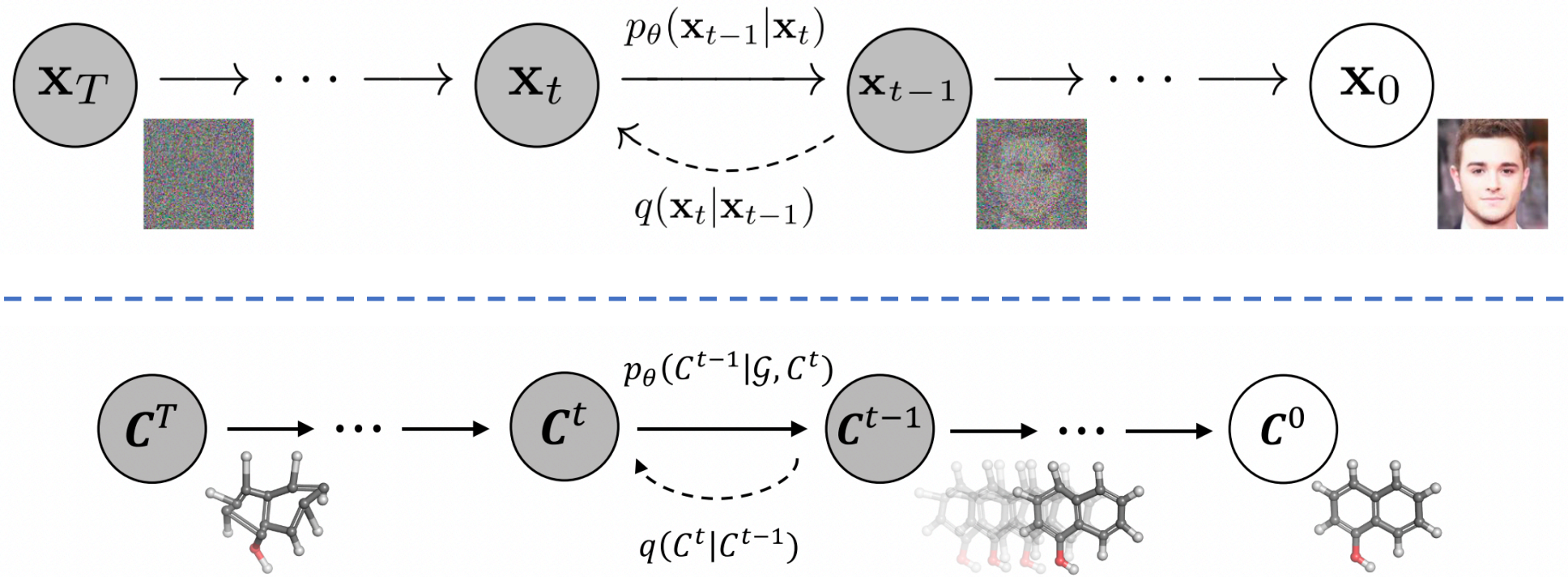


Figure 1: Illustration of the diffusion and reverse process of GEODIFF. For diffusion process, noise from fixed posterior distributions $q(\mathcal{C}^t|\mathcal{C}^{t-1})$ is gradually added until the conformation is destroyed. Symmetrically, for generative process, an initial state $\mathcal{C}^T$ is sampled from standard Gaussian distribution, and the conformation is progressively refined via the Markov kernels $p_\theta(\mathcal{C}^{t-1}|\mathcal{G}, \mathcal{C}^t)$.