International Journal of Applied Science; Vol. 8, No. 2; 2025
ISSN 2576-7240 E-ISSN 2576-7259
https://doi.org/10.30560/ijas.v8n2p117

Generative Graph based Model Inversion Attack on Graph Neural Network

Hongfa Ding[1], Tian Tian[1] & Shiyun He[1]

[1] School of Information, GuiZhou University of Finance and Economics, China

Correspondence: Hongfa Ding, School of information, Guizhou University of Finance and Economics, Guiyang, China. E-mail: hongfa.ding@gmail.com

Received: May 12, 2025; Accepted: May 29, 2025; Published: May 30, 2025

The research is financed by the Science and Technology Program of Guizhou(No. Qian Sci. Contr. Achiev. [2024]Major-17, Qian-Sci. Sontr. Platform ZSYS[2024]003), the Natural Science Researching Program of D.o.E. of Guizhou (No. Qian Edu. & Tech. [2024] 326, Qian Edu. & Tech. [2023] 065, Qian Edu. & Tech. [2023]014), and the Science and Technology Program of Guanshanhu District(No. Guan Science Contract [2023]16).

Abstract

Aiming at the privacy leakage risks of Graph Neural Networks (GNNs) in black-box scenarios, this paper proposes a Generation-Graph based Model Inversion Attack on GNN (GenG-MIA). By constructing a generative attack framework and integrating public knowledge distillation with structural optimization strategies, the proposed method effectively addresses challenges such as the high-dimensional sparsity of graph structure data, generative bias, and model collapse. GenG-MIA operates in two stages: first, during the public knowledge distillation stage, Wasserstein GAN is employed to train generators and discriminators on public datasets, enhancing the authenticity and diversity of generated graphs through a diversity loss term and introducing local/global discriminators to mitigate semantic gaps; second, in the structure revelation stage, potential vector projections are optimized to align with the feature space of the target model, thus recovering missing sensitive structures in training graphs. Experimental results show that GenG-MIA significantly outperforms existing methods in terms of attack accuracy and efficiency, enabling the efficient reconstruction of the topological structures of target training graphs and providing a new paradigm for privacy risk assessment of GNN models. This study further expands the application potential of generative attacks in complex graph data scenarios and offers theoretical references for privacy protection and model robustness design.

Keywords: graph neural network, model inversion attack, generative adversarial network, graph structure data

1. Introduction

With the remarkable success of machine learning and deep learning in multiple domains, recent studies have shown that the training phase of machine models involves large-scale training data, which often contains sensitive privacy information. During training, models may intentionally or unintentionally "memorize" information about the training data. Attackers can leverage this memorized information to launch privacy attacks, potentially leaking sensitive information in the training data and threatening user privacy. This has triggered typical data confidentiality issues, leading to the consideration that data protection rights and obligations may apply to models themselves, which has raised profound concerns about model security.

The model inversion attack was first proposed by (Fredrikson, 2014) for linear regression models, aiming to learn sensitive genomic information about individuals. Later, (Fredrikson, 2015) extended this concept to shallow neural networks to extract facial information. They treated model inversion as an optimization problem and solved it through gradient descent on images. This optimization-based approach aims to transform model inversion into a gradient-based optimization process without additional training of models for inversion tasks (Zhang, 2021; Liu, 2020; Duddu, 2020), but it is typically applied only in white-box settings, where the learning objective is optimized iteratively. Additionally, some researchers (Wang, 2021) first introduced model inversion attacks into collaborative networks to address inference data privacy issues. Under different settings, they used the intermediate outputs of neural networks to reconstruct input images and evaluated the attack on different models and datasets. Although the method (Wang, 2021) proposed an approximate calculation of mutual information, for high-dimensional data or complex models, approximation errors may affect attack effectiveness. Other researchers (Yin,

2023) argued that previous model inversion attacks only demonstrated the possibility of recovering input data with given gradients under very strict conditions and introduced GradeInversion (Yin, 2023), which customizes an optimization task to transform random noise into natural images to match large-batch gradients while regularizing image quality. However, this experiment only targeted low-resolution images, and for high-resolution inputs, information loss in intermediate features may be more significant, potentially causing a significant decline in the quality of reconstructed images.

2. Design of the GenG-MIA Model

This section proposes a Generation-Graph based Model Inversion Attack (GenG-MIA) on GNN. First, based on the acquisition of public graph datasets and posterior probabilities output by the target model, a graph inversion attack framework based on generative models is constructed. Public knowledge distillation is employed to train the generator, encouraging the authenticity and diversity of generated graph data. Second, the generator obtained from the previous step is utilized to recover missing sensitive structures in the training graph data, thereby significantly improving the accuracy and efficiency of the attack. On this basis, the framework can better simulate the behavior of the target model, enabling more effective attacks on GNN models under various application scenarios and attack conditions. Finally, experiments are conducted to validate the effectiveness and usability of the framework. By comparing with existing methods, the advantages of this study's approach in exposing privacy risks of GNN models and graph data using generative methods are highlighted. This chapter considers implementing generative graph inversion attacks under more realistic settings with minimal functional sets and further extends them to a general scenario attack framework.

2.1 Assumptions of the Threat Model

This paper assumes a threat model similar to existing model inversion attacks (Fredrikson, 2014). Specifically, in the black-box GNN scenario where the model only releases confidence values without disclosing model parameters, an adversary can leverage the adversarial generation capability of GANs to map knowledge such as confidence values, node attributes, and public datasets into topological structure generation priors, and reconstruct adjacency edges through adversarial training and iterative optimization. However, directly adapting generative model inversion attacks to graph structures leads to three key challenges:

1) The adjacency vectors of graph structure data are high-dimensional, discrete, and sparse, causing the generator to produce a large number of adjacency edges during generation and triggering the curse of dimensionality.

2) Prior knowledge easily introduces modeling bias. Briefly, public graph data, node features, and confidence values may contain irrelevant attributes (e.g., user height in social networks has little relevance to friend relationship prediction), creating a semantic gap with real graph structures and leading the generator to learn incorrect correlations.

3) GAN training is prone to model collapse. In other words, during graph generation, on one hand, specific subgraphs in the generated graph are heavily repeated; on the other hand, a large number of semantically irrelevant edges are generated.

2.2 Attacker's Knowledge and Capabilities

We focus on graph structure inversion under black-box settings. The attacker is assumed to have access to the target model and use inference techniques to recover the adjacency matrix of the target training graph. It is assumed that the attacker possesses all node labels and features. In addition to the target model and node labels, the attacker may also have other auxiliary knowledge, such as graph generative models, node attributes X , node IDs, or model output confidence values, to facilitate model inversion. The corresponding attack objective is to reconstruct the original training graph structure. We will discuss the impact of auxiliary knowledge and the number of node labels on attack performance in the following sections.

2.3 Attacker's Auxiliary Knowledge

Auxiliary knowledge can be the relational structure containing only a small part of the target graph data, which provides preliminary connection patterns and association clues between nodes in the target graph data. It can also be graph data of the same type. Taking the Cora citation graph dataset as an example, any other citation-type graph dataset can be used as auxiliary knowledge. Since graph data of the same type share similarities in data generation mechanisms, semantic expressions, and structural features, attackers can mine common information and transfer it to the target dataset. In public graph datasets, the relationships between nodes and edges are relatively stable. Attackers can use generative adversarial network (GAN) models to learn the relationship patterns and connection probabilities between nodes and edges in public graph datasets through adversarial training between generators and discriminators, and then deduce information about the target dataset. GenG-MIA mainly uses public data

distillation, which transfers rich knowledge from public graph datasets, such as complex structural information and node feature distributions, to student models through a teacher-student model architecture and specific loss function designs (e.g., soft-label loss, feature-matching loss). This allows student models to integrate public data knowledge while learning target graph data features, enhancing their defense capabilities against attacks based on such auxiliary knowledge and improving model security and stability in complex attack environments. Additionally, auxiliary datasets can also be other types of graph datasets. Although the structural and semantic relevance between these different types of graph datasets and the target graph data is relatively weak, in the absence of better auxiliary knowledge, attackers can still attempt to obtain some general graph structure features or node attribute features from them, though the attack effect based on such data may be worse than that of homogeneous graph data.

2.4 Attacker's Graph Generative Model

The graph generative model can be any graph generative model obtained from open-source platforms and deployed on the attacker's own platform, or it can be a graph generative model written by the attacker. Different graph generative models mainly differ in the scale and quality of graph data generated according to different strategic directions. Graph generative models mainly include generators and discriminators, which work together to generate high-quality graph data. GenG-MIA primarily focuses on generative models that can produce realistic graph structures.

2.5 The Framework of GenG-MIA

As shown in Figure 1, the proposed generative graph model inversion attack framework for reconstructing the topological structure of graph data operates by first training generative and discriminative models on public datasets to foster the creation of graph data with authenticity and diversity. Next, leveraging the generator trained in the initial stage, the framework recovers missing sensitive regions in the target graph data through solving optimization problems. Finally, the target classifier is used to classify the generated data, further differentiating between generated and real data to enable reconstruction of data from the target network's private training set.

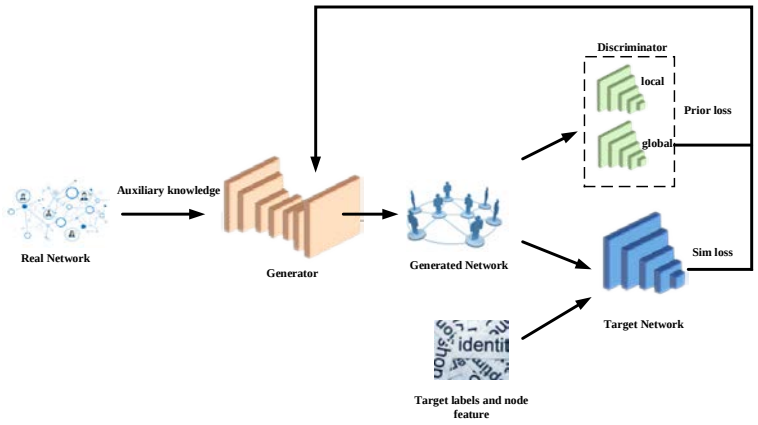

Figure 1. Flowchart of Generative Graph based Model Inversion Attack

To realistically reconstruct the missing graph structural relationships in graph data, generators and discriminators trained on a public training set are used to realistically reconstruct the missing sensitive regions in the training graph data. After training, the goal is to find the latent vector z that achieves the maximum likelihood under the target network while restricting it to the learned data flow manifold. However, without proper design, the generator may not allow the target network to easily distinguish between different latent vectors. For example, in an extreme case, if the graph data generated from all latent vectors converges to the same point in the feature space of the target network model, there is no hope of identifying which one is more likely to appear in the private training set of the target network. To address this issue, this chapter proposes a simple yet effective method, GenG-MIA. The reconstruction process of this method consists of two stages: (1) Public knowledge distillation, where the generator and discriminator are trained on a public dataset to encourage the authenticity and diversity of graph data generated by the generator. The public dataset can be of other types and has no class overlap with the private dataset. (2) Structure revelation, where the generator obtained in the first stage is utilized to recover the missing sensitive structure in the graph data by solving an optimization problem.

For the first stage, When auxiliary knowledge (such as part of the target graph structure or a version of the same-type graph dataset) is available to the attacker, this chapter takes the auxiliary knowledge as an additional input to the generator. Additionally, when the additional knowledge is from the same-type graph dataset, this chapter employs two discriminators to identify whether the generated graph data is real or artificial. The global discriminator reconstructs the global graph structure to assess its overall coherence, while the local discriminator ensures the local consistency of the graph structure. uses the classic Wasserstein GAN (Arjosky,2017) training loss:

$$\min_{G}\max_{D} L_{wgan}(G,D) = E_x[D(x)] - E_z[D(G(z))]$$

When auxiliary knowledge (such as part of the target graph structure or a version of the same-type graph dataset) is available to the attacker, this chapter takes the auxiliary knowledge as an additional input to the generator. Additionally, when the additional knowledge is from the same-type graph dataset, this chapter employs two discriminators to identify whether the generated graph data is real or artificial. The global discriminator reconstructs the global graph structure to assess its overall coherence, while the local discriminator ensures the local consistency of the graph structure.

Furthermore, this chapter introduces a diversity loss term that promotes the diversity of the synthetic graph data when $G(\cdot)$ is projected into the feature space of the target network. Let F denote the feature extractor of the target network. Therefore, the diversity loss can be expressed as:

$$\min_{A'} \alpha \cdot L_{sim}(A', X) + \beta \cdot L_{conf}(f_\theta, A', y) + \gamma \cdot R_{sparse}(A') + \delta \cdot L_{adv}(A')$$

$$s.t. \qquad A' = G(X;\Theta_G),\ A'_{ij} \in [0,1],\ \forall i, j$$

Where the feature similarity loss $L_{sim} = \| A' - XX^T \|_F^2$;the model confidence loss $L_{conf} = \| f_\theta(X, A') - f_\theta(X, A) \|_F^2$;the sparse term $R_{sparse} = \frac{1}{N^2}\sum_{i,j} A'_{ij}$ and the adversarial training loss $L_{adv} = -\mathrm{E}_{(u,v)\sim p(A')}[D_{local}(X,(u,v))]$ .As described above,greater diversity will help the target network identify the generated graph data that is most likely to appear in its private training set.

3. Results

To systematically evaluate the effectiveness of generative graph model inversion attacks on GNN models, this study selects multiple publicly available real - world datasets for experimental and comparative analysis. During the experiments, the effectiveness of the attacks is first evaluated using different datasets under a variety of evaluation metrics, with the experimental results then analyzed and verified. Next, the impact of public knowledge is experimentally verified. Finally, ablation experiments are conducted to analyze the contribution of each component. All experiments are implemented using the Python 3.9 programming language on an Ubuntu 18.04 LTS operating system, with a Hygon 7381 processor, 32GB of RAM, and a DCU Z100L 32G graphics card.

3.1 Experimental Setup

In this study, three highly representative real - world graph datasets are selected from the Stanford Large Network Dataset Collection website for experiments. These datasets possess distinct characteristics in terms of network structure and data features, providing rich data support for comprehensively evaluating the effectiveness of generative graph model inversion attacks in GNN models. The datasets are Cora, Citeseer, and Polblogs. Table 1 briefly analyzes the nodes, edges, and node features of the experimental datasets.

Table 1. Datasets

Datasets	$\|V\|$	$\|E\|$	X	*Classes*
Cora	2708	5278	1433	7
Citeseer	3327	4552	3703	6
Polblogs	1490	19025	-	2

1) Conventions: Each dataset is split into two disjoint parts: one as a private dataset for training the target network and the other as a public dataset for prior knowledge distillation. Throughout the experiments, there is no class overlap between the public data and the private training data of the target network. This ensures fairness by allowing the adversary to acquire only generic feature knowledge across all classes from the public dataset, without access to private, class - specific features used in training the target network.

2) Model: The target model is a Graph Convolutional Network (GCN) consisting of two convolutional layers and one fully connected layer, with parameters consistent with the original literature. The GCN was chosen as the experimental baseline due to its foundational role in GNNs. During training, 10% of nodes are randomly sampled as the training set, 20% as the validation set, and the model is trained for 200 epochs using a gradient strategy based on convergence behavior and accuracy.

3) Parameter Settings: In terms of model architecture, the hidden layer dimension is set as Hidden_dim=128, with the hidden layer dimension of the local discriminator being 32 and that of the global discriminator being 64. For training parameters, the generator undergoes 30,000 default training epochs with a learning rate of 0.0005, while the discriminator also uses a learning rate of 0.0005; the number of edges sampled each time and the attack training iteration count are specified according to experimental requirements. In the loss function configuration, the global discriminator loss weight is 0.7, the local discriminator loss weight is 0.3, the global deception loss weight is 0.7, the local deception loss weight is 0.3, the similarity loss (qu) weight is 1, the confidence loss (confidence) weight is -2.5, and the sparsity loss (sparse_loss) weight is 0.8.

4) Evaluation Metrics: Since the attack is unsupervised and the adversary cannot determine a specific threshold for predictions, this study follows prior model inversion work by using Area Under the Receiver Operating Characteristic Curve (AUC) and Average Precision (AP) as primary metrics. AUC and AP are computed using all edges from the training graph and an equal number of randomly sampled non - connected node pairs.

3.2 Performance Analysis

GenG-MIA is compared with the state-of-the-art gradient-based model inversion attack GraphMI (Zhang, 2021) and the early inverse-model-based attack GE (Zhang,2022). The black-box attack in RL-GraphMI (Zhang, 2021) is the first model inversion attack proposed for GNNs, where the adversary uses reinforcement learning for graph structure reconstruction and returns the graph structure with the closest label value distance. GE is one of the earliest attack algorithms on GNNs that uses the decoder of an autoencoder to reconstruct graphs from graph embeddings. The results of model inversion attacks on GCN models are summarized in Table 2. As shown, by controlling the generator's training iterations and time, GenG-MIA achieves balanced performance across almost all datasets. In contrast, optimization-based methods are highly unstable-gradient descent often leads to meaningless local minima. Existing model inversion attacks tend to generate adversarial samples that can deceive the target network but lack recognizable privacy features, making them more covert. Additionally, we observe that all methods achieve lower accuracy on the Polblogs dataset. As shown in Figures 2 and 3, which display the topological structures and degree distributions of each dataset, Cora and Citeseer have dense central clusters with skewed degree distributions (high influence in central nodes), while Polblogs has fewer central clusters, a more uniform degree distribution, and no node features-limited edge information inference likely contributes to the poor attack performance on this dataset.

Table 2. Performance Comparison of GCN Black-Box Attacks

Meth	Cora		Citeseer		Polblogs	
	AUC	AP	AUC	AP	AUC	AP
GE	0.745	0.722	0.765	0.77	0.59	0.564
RL-GraphMI	0.744	0.691	**0.871**	0.844	0.526	0.402
GenG-MIA	**0.796**	0.768	0.782	0.723	**0.652**	0.648

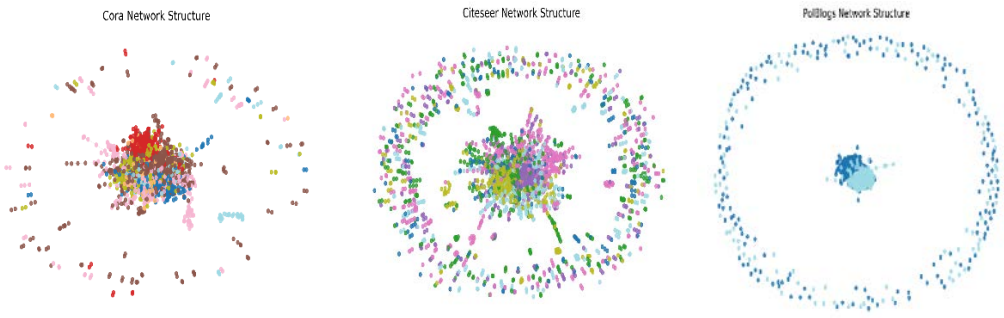

Figure 2. (a) Topological structure of the Cora dataset; (b) topological structure of the Citeseer dataset; (c) topological structure of the Polblogs dataset

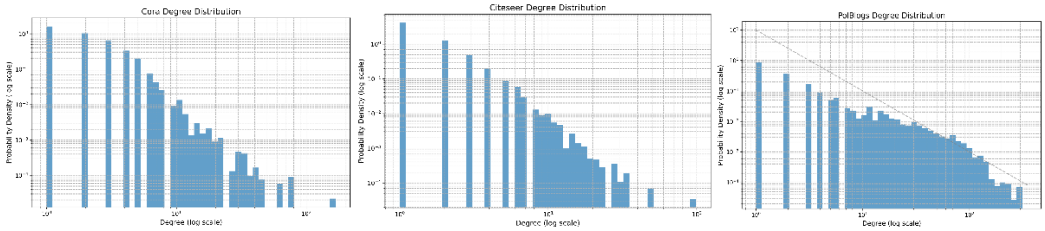

Figure 3. (a) Degree distribution of the Cora dataset; (b) degree distribution of the Citeseer dataset; (c) degree distribution of the Polblogs dataset

3.3 Ablation Experiments

To validate the effectiveness of key components in the GenG-MIA framework, we design systematic ablation experiments by progressively removing core modules to quantify performance changes and visually assess each component's contribution to attack utility. Experiments are conducted on the Cora dataset using the GCN model, with evaluation metrics including AUC and AP, and the generator trained for 10,000 iterations. As shown in Table 3, removing either the local or global discriminator leads to a decrease in AUC. Figure 4 illustrates the loss dynamics during training: removing the local discriminator (Figure 4a) causes unstable oscillations, while removing the global discriminator (Figure 4b) results in more stable but suboptimal training. This indicates that the local discriminator stabilizes training and enhances fine-grained recovery capabilities.

Table 3. Impact of Component Ablation on Attack Performance

Configuration	AUC	AP
Full GenG-MIA	0.796	0.768
W/o Local Discriminator	0.748	0.714
W/o Global Discriminator	0.732	0.726

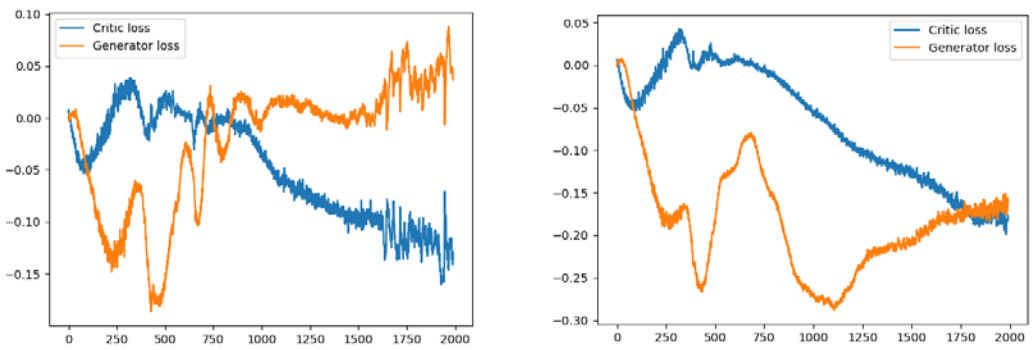

Figure 4. (a) Training loss without the local discriminator; (b) training loss without the global discriminator.

These results demonstrate that multi-scale discriminators collaboratively constrain the generation process from both macro-topological coherence and micro-structural consistency, mutually reinforcing each other. Generative samples lacking diversity loss exhibit severe model collapse in the feature space. Theoretical analysis shows that the diversity loss enforces linear independence of generated samples in the target model's feature space via orthogonal constraints, avoiding local optima during optimization.

4. Conclusion

This paper actively explores the privacy risks in GNN models within the context of model deployment, addressing the challenge of introducing GAN into graph model inversion attacks for GNNs for the first time. The generative graph model inversion attack, GenG-MIA, designs a hybrid similarity generator and dual-discriminator collaborative training under black-box settings, optimizes generator parameters through multi-objective loss training with alternating iterative updates, and controls sparsity and value ranges using regularization and masking matrices. By leveraging GANs, it reduces dependency on model queries and parameter control, balances high-order semantics with local topological consistency via a hybrid similarity generator, and overcomes the perception limitations of single discriminators through dual-discriminator collaboration. Based on this framework, the study proposes a generative attack algorithm to explore broader privacy risks in GNNs and improve attack efficiency. Experimental comparisons on multiple public real-world datasets-split into private and public subsets for fairness-show that increasing iterations enhance attack performance, with generated graph data approaching real data in target classifier performance, though this comes with significantly longer training times and a larger feature-space distance between generated and real data, necessitating trade-offs between attack performance and computational costs in practical applications.

References

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein generative adversarial networks. *Proceedings of the 34th International Conference on Machine Learning*, 70, 214–223. https://proceedings.mlr.press/v70/arjovsky17a.html

Duddu, V., Boutet, A., & Shejwalkar, V. (2020). Quantifying privacy leakage in graph embedding. *Proceedings of the 17th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous)*, 76–85. https://doi.org/10.1145/3448891.3448939

Fredrikson, M., Jha, S., & Ristenpart, T. (2015). Model inversion attacks that exploit confidence information and basic countermeasures. *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 1322–1333. https://doi.org/10.1145/2810103.2813677

Fredrikson, M., Lantz, E., Jha, S., Lin, S., Page, D., & Ristenpart, T. (2014). Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. *Proceedings of the 23rd USENIX Security Symposium*, 17–32. https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/fredrikson

Liu, X., Liu, K., Li, X., Su, J., Ge, Y., Wang, B., & Luo, J. (2020). An iterative multi-source mutual knowledge transfer framework for machine reading comprehension. *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI)*, 3794–3800. https://doi.org/10.24963/ijcai.2020/525

Wang, T., Zhang, Y., & Jia, R. (2021). Improving robustness to model inversion attacks via mutual information regularization. *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, 35(13), 11666–11673. https://doi.org/10.1609/aaai.v35i13.17394

Yin, Y., Zhang, X., Zhang, H., Wang, Y., & Chen, X. (2023). Ginver: Generative model inversion attacks against collaborative inference. *Proceedings of the 32nd USENIX Security Symposium*, 2122–2131. https://doi.org/10.48550/arXiv.2302.12345

Zhang, Z., Chen, M., Backes, M., Shen, Y., & Zhang, Y. (2022). Inference attacks against graph neural networks. *Proceedings of the 31st USENIX Security Symposium*, 4543–4560. https://doi.org/10.48550/arXiv.2110.02631

Zhang, Z., Liu, Q., Huang, Z., Wang, H., Lee, C.-K., & Chen, E. (2022). Model inversion attacks against graph neural networks. *Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI)*, 35(9), 8729–8741. https://doi.org/10.24963/ijcai.2022/121

Zhang, Z., Liu, Q., Huang, Z., Wang, H., Lu, C., Liu, C., & Chen, E. (2021). GraphMI: Extracting private graph data from graph neural networks. *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI)*, 3749–3755. https://doi.org/10.24963/ijcai.2021/516