Unveiling Privacy Vulnerabilities: Investigating the Role of Structure in Graph Data

Hanyang Yuan*
Zhejiang University
Hangzhou, China
Fudan University
Shanghai, China
yuanhy0408@zju.edu.cn

Jiarong Xu†
Fudan University
Shanghai, China
jiarongxu@fudan.edu.cn

Cong Wang
Peking University
Beijing, China
wangcong@gsm.pku.edu.cn

Ziqi Yang
Zhejiang University
Hangzhou, China
yangziqi@zju.edu.cn

Chunping Wang
Finvolution Group
Shanghai, China
wangchunping02@xinye.com

Keting Yin
Zhejiang University
Hangzhou, China
yinkt@zju.edu.cn

Yang Yang
Zhejiang University
Hangzhou, China
yangya@zju.edu.cn

ABSTRACT

The public sharing of user information opens the door for adversaries to infer private data, leading to privacy breaches and facilitating malicious activities. While numerous studies have concentrated on privacy leakage via public user attributes, the threats associated with the exposure of user relationships, particularly through network structure, are often neglected. This study aims to fill this critical gap by advancing the understanding and protection against privacy risks emanating from network structure, moving beyond direct connections with neighbors to include the broader implications of indirect network structural patterns. To achieve this, we first investigate the problem of Graph Privacy Leakage via Structure (GPS), and introduce a novel measure, the Generalized Homophily Ratio, to quantify the various mechanisms contributing to privacy breach risks in GPS. Based on this insight, we develop a novel graph private attribute inference attack, which acts as a pivotal tool for evaluating the potential for privacy leakage through network structures under worst-case scenarios. To protect users' private data from such vulnerabilities, we propose a graph data publishing method incorporating a learnable graph sampling technique, effectively transforming the original graph into a privacy-preserving version. Extensive experiments demonstrate that our attack model poses a significant threat to user privacy, and our graph data publishing method successfully achieves the optimal privacy-utility trade-off compared to baselines.

CCS CONCEPTS

• **Security and privacy → Social network security and privacy**.

KEYWORDS

Graph privacy protection, data release, adversarial learning

ACM Reference Format:
Hanyang Yuan, Jiarong Xu, Cong Wang, Ziqi Yang, Chunping Wang, Keting Yin, and Yang Yang. 2024. Unveiling Privacy Vulnerabilities: Investigating the Role of Structure in Graph Data. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24), August 25–29, 2024, Barcelona, Spain.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3637528.3672013

1 INTRODUCTION

In the era of big data, with the increasing involvement of personal data in information technology and shared on the web, privacy protection has emerged as a crucial concern [21, 24, 41]. In real-world scenarios, individuals often share some information publicly while safeguarding their private attributes. However, publicly available user information gives adversaries opportunities to infer private attributes, resulting in privacy breaches [40, 42, 44]. Furthermore, the inferred private data can facilitate malicious activities. For instance, in the 2010s, Cambridge Analytica collected personal data from 87 million Facebook users to infer their political stands, which were further used for political advertising, resulting in a scandal with over $100 billion in economic losses [20]. This incident emphasizes the urgent need for privacy protection mechanisms.

To prevent the exposure of user privacy, traditional works typically focus on the privacy leakage through users' public attributes [21,

*State Key Laboratory of Blockchain and Security. The author is also at Hangzhou High-Tech Zone (Binjiang) Institute of Blockchain and Data Security, Hangzhou, China. This work was done when the author was a visiting student at Fudan University.
†Corresponding author.

KDD '24, August 25–29, 2024, Barcelona, Spain

ACM ISBN 979-8-4007-0490-1/24/08…$15.00
https://doi.org/10.1145/3637528.3672013

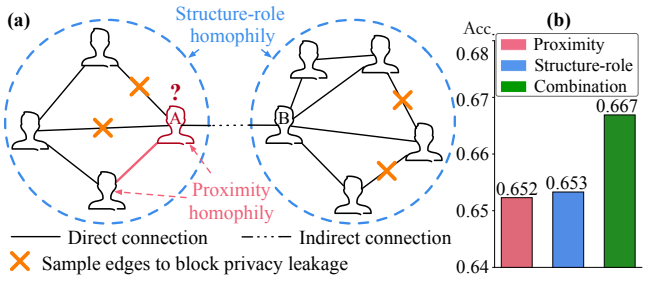


Figure 1: (a) Illustration of privacy leakage mechanisms: proximity homophily highlighted in pink, structure-role homophily in blue, alongside the privacy protection strategy depicted in orange. (b) The results of private attribute inference attacks accounting for proximity homophily, structure-role homophily, and a combination of both on Pokec-n.

47, 48]. However, these methods often overlook the privacy risks stemming from the public relationships among users [22, 40, 59]. For example, in an online social platform like Facebook, users often publicly display their followers or friends, where these relationships collectively form a network. This network structure can also give rise to potential privacy leakage [22, 40, 59].

This work delves into the problem of Graph Privacy leakage via Structure (GPS), aimed at unveiling various mechanisms by which network structure can lead to privacy exposure. Extant works on privacy breach through network structure are primarily premised in social homophily theory [39] that posits users with similar private attributes tend to connect with each other. Hence, they study privacy leak through direct connections with neighbors [17, 22, 40, 59]. However, user privacy in networks can also be compromised through more complex structural patterns, extending beyond direct neighbors. Figure 1 (a) illustrates that privacy risks for user "A" arise not only from direct neighbors (*i.e.*, proximity information), but also from users like "B" who, despite not being direct connections, exhibit similar local structures (*i.e.*, structure-role information). An example of structure-role information in social networks is the observation that younger and older users tend to maintain social circles of different sizes [12]. This dimension of privacy leakage, facilitated by such local structures, has not been thoroughly investigated in existing studies. To fill this research gap, we aim to develop *a graph data publishing method aimed at comprehensively protecting against potential privacy breaches arising from the network structure.* Nevertheless, achieving this goal presents several challenges.

The first challenge lies in how to measure the extent of privacy exposure through network structures. Previous research mainly focus on privacy leakage through direct connections between nodes, using homophily to quantify proximity-related exposure [23, 35, 49]. However, this approach falls short in assessing privacy risks from structure-role information. In this study, we introduce the Generalized Homophily Ratio (GHRatio), a novel measure to quantify privacy risks associated with network structures. The GHRatio is a general form that is adaptable to various structural features. We explore two prevalent cases—proximity homophily, structure-role homophily and their combination— that contribute to privacy leakage.

The second challenge stems from the necessity to develop a private attribute inference attack model that utilizes proximity homophily, structure-role homophily, and their combination to launch attacks. Given that existing attack strategies merely exploit proximity homophily [3, 22, 40], suboptimal results are yielded (as depicted by the red bar in Figure 1 (b)). To overcome this challenge, our model is designed to account for all identified privacy breaches through a data-centric strategy. This strategy involves providing a graph neural network (GNN) with various data forms, thus enhancing its capacity to learn from different types of homophily. Consequently, our attack model effectively behaves like a worst-case adversary, as evidenced by the green bar in Figure 1 (b).

The last challenge lies in how to design a graph data publishing approach that can effectively defend the worst-case private attribute inference attack. Previous efforts in graph data publishing have primarily focused on differential privacy (DP) [57, 60] and graph sampling [5, 17], but DP often compromises the utility of the data [30]. In addition, many sampling methods are rule-based and reliant on domain-specific knowledge [37, 38], which restricts their applicability. We therefore propose a learnable graph sampling method for privacy protection, employing a generative network that selectively samples edges to block privacy leakage (as illustrated by ✕ in Figure 1 (a)). This method ultimately produces a sampled graph suitable for publication.

Our contributions are summarized as follows:

- **Problem and measure**: Our work pioneers a comprehensive investigation into the problem of Graph Privacy leakage via Structure (GPS), introducing the innovative Generalized Homophily Ratio (GHRatio) as a measure of privacy leakage. This helps us unveil all identified mechanisms by which the network structure can lead to privacy breaches in a quantitative manner.
- **Attack model**: We introduce a novel private attribute inference attack leveraging a data-centric strategy to exploit all identified privacy breaches. By feeding a GNN various data forms, it gains the ability to learn from multiple homophily types that result in privacy risks.
- **Defensive model**: To counter the attacks, we propose a graph data publishing method that employs learnable graph sampling, rendering the sampled graph suitable for publication.
- **Extensive experiments**: Experiments in five real-world scenarios demonstrate that (1) our private attribute inference attack beats the best baselines by an average of +2.93%, and (2) our graph data publishing approach achieves the optimal privacy-utility trade-off, outperforming existing defensive methods when evaluated against worst-case attack scenarios.

2 PROBLEM DEFINITION

Network structure is a significant factor contributing to privacy breaches, which stands as a fundamental data source for various network analysis tasks, such as node classification and community detection. Previous research has proven that modifying network

structure is more effective than modifying node attributes in enhancing privacy protection [17]. Therefore, this work particularly focuses on a privacy-preserving graph data publishing problem with a specific emphasis on the network structure.

In this study, we refer to privacy as a particular attribute that nodes choose to keep hidden, which aligns with previous works [17, 28]. Let graph $G = (V, E, X)$ denote an undirected network, where $V = \{v_1, ..., v_n\}$ is the node set, $E \subseteq V \times V$ is the edge set, and $X \in \mathbb{R}^{n \times m}$ is the node attribute matrix. $A \in \mathbb{R}^{n \times n}$ is the adjacency matrix of G, where $A_{ij} = 1$ if there exists an edge $(i, j) \in E$, otherwise $A_{ij} = 0$. Each node $v_i \in V$ is associated with a known/unknown private attribute Z_i. Here, $Z_i \in Z = Z_L \cup Z_U$, where Z_L denotes publicly available private attributes, and Z_U represents hidden private attributes. In this context, privacy in the graph G is defined by the set of hidden private attributes Z_U.

Let us consider the following attack scenario. In a public social network, some users choose to conceal their private attributes, while others make them public. The adversary aims to infer these hidden private attributes. The adversary is assumed to have access to the network structure, node attributes (typically non-private), and publicly available private attributes. Publicly available private attributes can come from users who do not consider this information private or who seek to maximize visibility by sharing extensive personal information.

Formally, we define this as graph private attribute inference attack problem.

Problem 1 (**Graph Private Attribute Inference Attack**). *Given graph $G = (V, E, X)$ and the publicly available private attributes Z_L, the graph private attribute inference attack aims to learn a function*

$$f : G, Z_L \rightarrow Z_U, \tag{1}$$

that predicts the hidden private attribute Z_U.

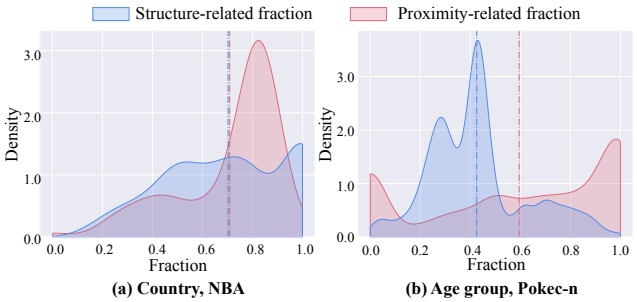
The primary objective of this work is to tackle the problem of privacy-preserving graph data publishing against the aforementioned graph private attribute inference attack. Specifically, instead of directly releasing the original graph, the data publisher is encouraged to generate a sampled graph for publishing, such that the sampled graph can defend against graph private attribute inference attack. Based on the above definition, we formulate our privacy-preserving graph data publishing problem as follows.

Problem 2 (**Privacy-preserving graph data publishing**). *Given graph $G = (V, E, X)$ and the publicly available private attributes Z_L, the data publisher aims to sample a new graph $G' = (V, E', X)$ by selectively removing edges in E, resulting in a new edge set E'. The sampled graph G' is expected to simultaneously achieve the following two objectives:*

Objective 1: privacy preservation. *The adversary with G' and Z_L cannot accurately infer the private attribute Z_U,* i.e.,

$$\min_{G'} \quad \text{perf}\left(f(G'), Z_U\right), \tag{2}$$

where perf *denotes a performance metric to evaluate how well the predicted value $f(G', Z_L)$ aligns with the ground truth Z_U, such as accuracy or ROC-AUC as used in our work.*



(a) Country, NBA **(b) Age group, Pokec-n**

Figure 2: Visualization of proximity-related fraction and structure-related fraction distributions on NBA and Pokec-n.

Objective 2: utility. *The sampled graph G' should not deviate too much from the original graph G. This ensures that G' conveys useful information.*

3 GRAPH PRIVACY-LEAKAGE VIA STRUCTURE (GPS)

In this section, we delve into the problem of GPS, examining how the structure of a graph can potentially lead to privacy breaches. We aim to introduce a novel measure to quantify the various mechanisms contributing to privacy breach risks in GPS.

We start with an exploratory analysis using the NBA and Pokec-n datasets, two widely adopted datasets for graph privacy-preserving learning [8, 9, 33]. We calculate (1) proximity-related fraction: the fraction of a node's neighbors sharing the same private attribute; and (2) structure-related fraction: the fraction of nodes with similar local structures and the same private attributes to the total nodes with similar local structures for a specific node. Here, similar local structures are defined where the structural similarity between nodes' ego networks exceeds a set threshold, assessed using degree centrality. From the results shown in Figure 2, we derive two key observations:

Observation 1: The necessity for node-level analysis. In Figure 2 (a), despite similar mean values for both fractions, their local distributions differ. Figure 2 (b) shows that even when the mean of the proximity-related fraction dominates, some nodes have significantly lower values of the proximity-related fraction. This challenges the graph-level homophily ratios from prior studies, which rely on mean values as indicators [32, 61], underscoring the need for node-level analysis.

Observation 2: Proximity, structure, and their combination should be simultaneously considered. We find that in Figure 2 (a), there are nodes in which both fractions are relatively high; and in Figure 2 (b), there are instances where the proximity-related fractions of certain nodes are small, while the structure-related fraction may provide supplementary information. These observations emphasize the importance of simultaneously considering proximity, structure, and their combination when addressing privacy concerns.

In light of the insights gained from the exploratory analysis, we propose a novel measure known as the Generalized Homophily

Ratio (GHRatio), which is a generalized form that can be used in conjunction with different definitions of structural features associated with graph privacy. Subsequently, we instantiate three forms of GHRatio: proximity homophily, structure-role homophily, and their combination. They represent the main pathways in graph structure through which privacy can be leaked.

Generalized Homophily Ratio. As our goal is to investigate how network structure discloses privacy, we begin by defining homophily indicator, which can be used to characterize the structural characteristic or relation between two nodes.

Definition 1 (Homophily indicator). *A homophily indicator $\delta(v_i, v_j, r)$ assesses whether node v_i and v_j exhibit a shared structural characteristic or relation r. For example, when r signifies similar local structure, we have $\delta(v_i, v_j, r) = 1$ if the similarity of v_i and v_j exceeds a certain threshold, and 0 otherwise; when considering r as the relation of adjacency, $\delta(v_i, v_j, r) = 1$ if node v_i and v_j are directly connected, and 0 otherwise.*

Based on the homophily indicator, we can define the GHRatio as follows.

Definition 2 (Generalized Homophily Ratio (GHRatio)). *Given an observed graph G and private attribute matrix Z, the GHRatio of node v_i is defined as the conditional probability that node v_i and any node v_j $(j \neq i)$ have the same private attributes, given that $\delta(v_i, v_j, r) = 1$,* i.e.,

$$\text{GHRatio}_i = P(Z_i = Z_j | \delta(v_i, v_j, r) = 1), j \neq i, \tag{3}$$

where Z_i and Z_j are private attributes of node v_i and v_j, respectively.

Two prevalent cases of GHRatio. Given the general form of GHRatio, we further delve into two prevalent cases of it: proximity homophily, structure-role homophily, by defining specific graph structural feature associated with GHRatio.

(1) Proximity homophily ratio. By defining the r in homophily indicator $\delta(v_i, v_j, r)$ as the relation of adjacency, we have $\delta(v_i, v_j, r) = 1$ if v_i and v_j are connected, and 0 otherwise; We name this specific case of GHRatio as Proximity Homophily Ratio ($\text{GHRatio}^{\text{prox}}$):

$$\text{GHRatio}_i^{\text{prox}} = \frac{|\{j | j \in \mathcal{N}(i) \wedge Z_j = Z_i\}|}{|\{j | j \in \mathcal{N}(i)\}|}, \tag{4}$$

where $\mathcal{N}(i)$ denotes v_i's neighborhood, and $|\cdot|$ denotes the cardinality of a set. In fact, $\text{GHRatio}^{\text{prox}}$ aligns with the node-level homophily ratio defined in existing works [35, 49].

(2) Structure-role homophily ratio. We define r in the homophily indicator $\delta(v_i, v_j, r)$ as the similar local structure. Then, we have $\delta(v_i, v_j, r) = 1$ if the similarity between the local structures of v_i and v_j exceeds a certain threshold, and 0 otherwise. We name this case of GHRatio as Structure-Role Homophily Ratio ($\text{GHRatio}^{\text{role}}$):

$$\text{GHRatio}_i^{\text{role}} = \frac{|\{j | g(j) \sim g(i) \wedge Z_j = Z_i\}|}{|\{j | g(j) \sim g(i)\}|}, \tag{5}$$

where $g(i)$ is the ego network of node v_i, $g(i) \sim g(j)$ denotes that the ego networks $g(i)$ and $g(j)$ are sufficiently similar (*e.g.*, this similarity can be understood in terms of structural similarity, specifically when the structural similarity between $g(i)$ and $g(j)$ exceeds a predetermined threshold). Empirically, we adopt the degree centrality to characterize the similarity between ego networks, which have been validated for its strong effectiveness and efficiency in previous studies [45, 58].

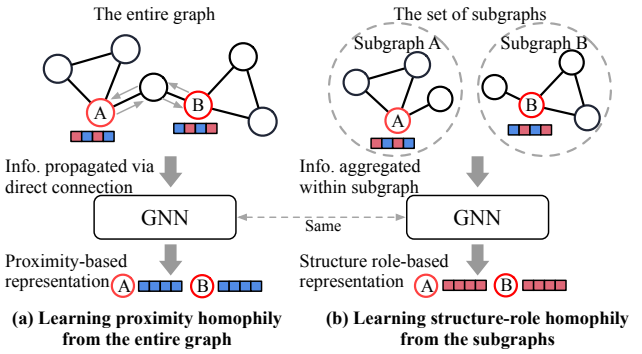


Figure 3: Illustration of our data-centric strategy of feeding different data forms (*i.e.*, graph vs subgraphs) into GNN to learn different knowledge (*i.e.*, proximity homophily vs structure-role homophily).

4 PRIVATE ATTRIBUTE INFERENCE ATTACK

This section introduces a novel private attribute inference attack model that leverages proximity homophily, structure-role homophily and their combination to disclose private information. We adopt a data-centric approach, feeding varied data forms into a GNN to extract the representations related to different homophily types (see § 4.1). Then, a routing operator is introduced for the adaptive integration of these homophily-related representations (see § 4.2). See Figure 4 (a) for an overview of our proposed attribute inference attack.

4.1 Enhancing GNNs with Data-Centric Strategy

Developing a GNN model capable of learning representations tailored to different homophily types is challenging. Existing works predominantly learn representations based on proximity information [15, 26] or high-order node dependencies [1, 61], which can not adequately learn structure-role information. Although some network representation methods [2, 16, 45] are designed to learn from node structure roles, their expressive power is limited.

In this work, we introduce a data-centric strategy designed to enhance GNNs' capacity to learn representations tailored to proximity homophily and structure-role homophily. This is achieved by feeding different forms of data into GNNs. An illustrative example is provided in Figure 3. Our key insight is:

(1) Feeding the entire graph to a GNN enables it to learn proximity homophily;
(2) Feeding the set of nodes' subgraphs (*e.g.*, ego networks) to a GNN, where each node's representation is computed as its subgraph's representation, facilitates the learning of structure-role homophily.

Learning proximity homophily from the entire graph. This principle aligns with the message-passing mechanism in standard

GNNs, where nodes update their representations by aggregating information from their direct neighbors [26, 53]. This aggregation process naturally encourages similarity in the representations of connected nodes [53]. Specifically, we generate proximity-based representations $H^{\text{prox}} \in \mathbb{R}^{n \times d}$ by feeding the entire graph G to a GNN encoder GNN^{prox}, *i.e.*,

$$H^{\text{prox}} = \text{GNN}^{\text{prox}}(G). \tag{6}$$

Learning structure-role homophily from subgraphs. This approach first extracts k-hop ego networks for each node, resulting in a set of subgraphs. These subgraphs are subsequently fed to a GNN encoder. The resulting subgraph representation, which encapsulates the aggregated features of its constituent nodes, serves as the representation for the subgraph's central node.

This benefits the GNN's ability to learn structure-role homophily for two main reasons: First, by concentrating on subgraphs, the central node's representation becomes exclusively reflective of its local structural context, isolating the central node from external influences of nodes outside the subgraph. Secondly, GNNs are particularly adept at learning structural information of smaller subgraphs. Although GNNs can identify structural patterns, comprehending complex structures in larger graphs remains challenging, as observed in [19]. Ultimately, this process allows the GNN to bring the representations of nodes with similar local structures closer in the latent space.

Specifically, we process the k-hop ego network of node i, denoted as $S_i = (V_i, A_i, X_i)$, and feed the subgraph to a GNN encoder GNN^{role}. The node representation H_i^{role} is computed as the subgraph representation, which is the mean of the representations of all nodes in the subgraph:

$$H_i^{\text{role}} = \text{Pooling}(\text{GNN}^{\text{role}}(S_i)), \tag{7}$$

where Pooling is the mean pooling operator.

Theoretical analysis of learning from subgraphs. While capturing proximity homophily from the entire graph is widely acknowledged [25, 53, 56], few works explore learning structure-role homophily from subgraphs. We therefore theoretically investigate whether GNNs fed with subgraphs can learn structure-role homophily. The following theorem suggests that nodes with similar local structures can obtain similar node representations.

Theorem 1. *Let S_i and S_j be two k-hop subgraphs induced from node v_i and v_j. After employing a K-layer GNN encoder with a 1-hop graph filter $\Psi(\mathcal{L})$ on each subgraph, the representations of the center node v_i and v_j are obtained via a pooling function,* i.e., *$H_i^{\text{role}} = \text{Pooling}(\text{GNN}^{\text{role}}(S_i))$ and $H_j^{\text{role}} = \text{Pooling}(\text{GNN}^{\text{role}}(S_j))$. Without loss of generality, assume that the attribute of each node is a vector of ones, H_i^{role} and H_j^{role} satisfy:*

$$\|H_i^{\text{role}} - H_j^{\text{role}}\|_2 \leq \tau \|\mathcal{L}_i - \mathcal{L}_j\|_2,$$

where $\|\cdot\|_2$ denotes L_2 norm of matrix or vector, τ denotes a constant depending on GNN^{role}, $\mathcal{L}_i$ denotes the normalised Laplacian matrix of S_i.

In Theorem 1, the term $\|\mathcal{L}_i - \mathcal{L}_j\|_2$ measures the difference of local structure around v_i and v_j. As similar local structures bring smaller differences in Laplacian matrices, the upper bound of node representation distance is reduced. Consequently, nodes with similar local structures become closer in the latent space. The proof of this theorem can be found in Appendix A.2.

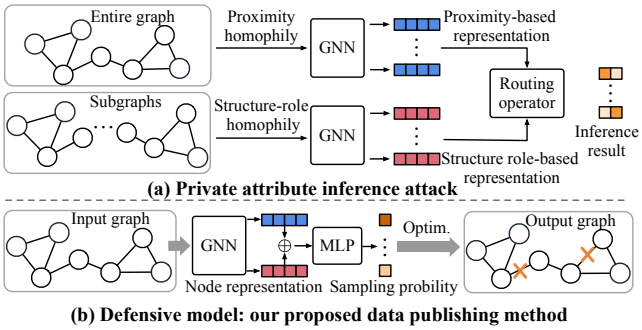


Figure 4: An overview of (a) the proposed attribute inference attack and (b) the proposed graph data publishing method.

4.2 Routing Operator

After obtaining the two types of representations, the challenge arises in determining the optimal method to integrate them, especially given the uncertainty about the extent of information that should be merged from each type. To tackle this issue, we introduce a *routing operator* that leverages our proposed GHRatio to effectively combine these representations.

Since $\text{GHRatio}^{\text{prox}}$ and $\text{GHRatio}^{\text{role}}$ serve to quantify the extent to which proximity and structural role disclose privacy, respectively. Utilizing these ratios, we can integrate the two types of representations, aiming for a balanced and informed combination that reflects the significance of both proximity and structural information in revealing privacy. However, calculating these ratios requires the knowledge of the private attributes of all nodes, which presents another difficulty. To address this, our approach involves estimating the ratios by employing the pseudo-labels of private attributes.

Formally, we apply Multilayer Perceptrons (MLPs) to obtain inference results from proximity-based representation H_{prox} and structure role-based representation H_{role}. Then we use the estimated GHRatios as the proportions to integrate them. The integrated result $\hat{Z}_i$ in turn serves as the pseudo-labels. Due to the interdependence between pseudo-labels and GHRatios, we initialize GHRatios with constants and iteratively update them during training. The above process can be described as follows:

$$\begin{aligned} &\hat{Z}_i^{\text{prox}} = \text{MLP}(H_i^{\text{prox}}), \hat{Z}_i^{\text{role}} = \text{MLP}(H_i^{\text{role}}), \\ &\hat{Z}_i = \text{GHRatio}_i^{\text{prox}} \cdot \hat{Z}_i^{\text{prox}} + \text{GHRatio}_i^{\text{role}} \cdot \hat{Z}_i^{\text{role}}, \end{aligned} \tag{8}$$

where $\hat{Z}_i$ denotes the final inference result for node v_i.

The model is optimized as follows:

$$\min_{\Theta} \quad L_{\text{adv}} = -\frac{1}{|V_L|} \sum_{v_i \in V_L} \sum_{c=1}^{C} Z_{i,c} \log \hat{Z}_{i,c}, \tag{9}$$

where C denotes the number of categories for the private attribute, V_L denotes the nodes with known private attributes, and Θ denotes the parameters of the attack model.

5 PRIVACY-PRESERVING GRAPH DATA PUBLISHING

In this section, we introduce a learnable graph sampling approach for privacy-preserving graph data publishing. We first outline the learnable strategy for graph sampling in § 5.1. Following this, the optimization objectives and training algorithm are detailed in § 5.2 and § 5.3, respectively. Figure 4 (b) provides an overview of the proposed graph data publishing method.

5.1 Learnable Graph Sampling

To generate a sampled graph G' with adjacency matrix A', we first feed the graph G into a GNN encoder to obtain node representation H^{samp}:

$$H^{\text{samp}} = \text{GNN}(G). \tag{10}$$

Given any connected node pair $\{v_i, v_j\}$, an MLP with sigmoid activation takes the concatenation of their node representations H_i^{samp} and H_j^{samp} as input to compute the probability $\mathcal{T}_{ij}$ of preserving the edge between $\{v_i, v_j\}$:

$$\mathcal{T}_{ij} = \sigma(\text{MLP}(H_i^{\text{samp}} \oplus H_j^{\text{samp}})), \tag{11}$$

where σ denotes sigmoid activation, and $\oplus$ represent the concatenation operation. With an edge's $\mathcal{T}_{ij}$ calculated, we sample whether to retain the edge in the synthetic graph G' according to this probability, where

$$A'_{ij} \sim \text{Bernoulli}(\mathcal{T}_{ij}). \tag{12}$$

In particular, the Gumbel-Softmax reparameterization trick [36] is utilized to tackle the non-differentiable nature of the sampling process. In doing so, we obtain a continuous sampling result, *i.e.*, $A'_{ij} = \sigma((\log U - \log(1-U) + \log \mathcal{T}_{ij})/\varepsilon)$, where $U \sim \text{Uniform}(0,1)$. As the temperature hyper-parameter ε tends to zero, the reparameterized result smoothly converges to binary values, while maintaining the relative order of each Gumbel [36].

5.2 Optimization Problem

We propose three optimization objectives for training learnable parameters within the graph sampling procedure: one aimed at defending against worst-case attacks, one designed for a broader spectrum of attacks, and another dedicated to preserving essential graph properties. These objectives collectively ensure that the sampled synthetic graph maintains user privacy while simultaneously achieving desirable data utility.

Defending Against Worst-Case Attack. Given the proposed inference attack, the most straightforward and effective approach is to defend against this attack under the worst-case. Specifically, to obtain the worst-case attack, we maximize the performance of the attack model, and subsequently, we defend against such an attack. This can be formulated as

$$\min_{\Phi} \max_{\Theta} \quad -L_{\text{adv}}(G'(\Phi), \Theta), \tag{13}$$

where Φ and Θ denote the parameters of the sampling component and the attack model respectively. Φ determines G' by influencing A'.

Defending Against a Broad Spectrum of Attacks via GHRatio. Eq. (13) ensures that our publishing method can defend against the proposed worst-case attack. However, in real-world scenarios, the released graph may face various attacks, and not all of them necessarily reach the worst case [34, 43]. In such scenarios, we devise a universal protection strategy via GHRatio, which serves as a measure independent of a specific attack.

Essentially, GHRatio quantifies how much information the network structure can disclose for inferring the private attribute. By minimizing GHRatio, we can mitigate the risk of privacy leakage in an attack-agnostic manner:

$$\min_{\Phi} \quad L_{\text{dis}} = \frac{1}{|V|} \sum_{v_i \in V} \|\text{GHRatio}_i(\Phi) - \text{GHRatio}_i^0\|, \tag{14}$$

where $\text{GHRatio}_i(\Phi)$ represents the new GHRatio of the sampled graph G'. GHRatio_i^0 represents $P(Z_i = Z_j)$, indicating the probability that nodes v_i and any node $v_j (j \neq i)$ have the same private attribute. When $L_{\text{dis}} = 0$, the structural characteristic in $\text{GHRatio}_i(\Phi)$ provides no benefits for attribute inference.

Note that since Z_i is given, we have $\text{GHRatio}_i^0 = P(Z_i)$. In practice, we propose to optimize the two prevalent cases of GHRatio, $\text{GHRatio}^{\text{prox}}$ and $\text{GHRatio}^{\text{role}}$, being described as:

$$\begin{aligned} \min_{\Phi} \quad L_{\text{dis}} = \frac{1}{|V|} \sum_{v_i \in V} & \|\text{GHRatio}_i^{\text{prox}}(\Phi) - \hat{P}(Z_i)\| \\ & + \|\text{GHRatio}_i^{\text{role}}(\Phi) - \hat{P}(Z_i)\|, \end{aligned} \tag{15}$$

where $\hat{P}(Z_i)$ represents the empirical estimation of $P(Z_i)$.

Regularization for Ensuring Utility. To ensure the utility of the sampled graph, we aim to align the properties of G' with those of G. Thus, we incorporate a reconstruction-based regularization term to control the deviation of the sampled graph:

$$\min_{\Phi} \quad L_{\text{reg}} = -\frac{1}{|E|} \sum_{(i,j) \in E} A_{ij} \log(\mathcal{T}_{ij}), \tag{16}$$

where $\mathcal{T}_{ij}$ denotes the sampling probability of edge (i, j). By Eq. (16), G' will retain as many edges as possible from G.

To sum up, we formalize the overall optimization problem:

$$\min_{\Phi} \max_{\Theta} \quad L = -\gamma \cdot L_{\text{adv}} + \eta \cdot L_{\text{dis}} + \lambda \cdot L_{\text{reg}}, \tag{17}$$

where $\gamma, \eta, \lambda > 0$ are hyper-parameters.

For different scenarios, we can also modify Eq. (17) to obtain different variants. If the goal is to protect against the proposed worst-case attack, only retaining L_{adv} and L_{reg} would be sufficient. On the other hand, if the goal is not specifically for the worst case but to be effective against a broad range of attacks, retaining L_{dis} and L_{reg} is suitable.

5.3 Training Algorithm

In the training phase, the parameters Φ of the sampling component and the parameters Θ of the proposed attack model are jointly trained. Specifically, The training algorithm iterates through the following main steps: (1) Learn Φ to minimize $-L_{\text{adv}}$, L_{reg} and L_{dis} while keeping Θ fixed, and (2) Learn Θ to maximize $-L_{\text{adv}}$ while keeping Φ fixed. Repeat these steps until the maximum iteration is reached. The detailed training algorithm and complexity analysis are summarized in Appendix A.3.

6 EXPERIMENTS

In this section, we evaluate the effectiveness of both the proposed attack model and the defensive model of data publishing.

6.1 Experiment Setting

Datasets. We conduct attribute inference and data publishing experiments on three datasets: Pokec-n, Pokec-z, and NBA [8, 33]. Following prior works [8], we treat country as the private attribute in NBA, and region as the private attribute in Pokec-n and Pokec-z. Additionally, we also treat users' age as another private attribute in Pokec-n and Pokec-z, categorizing it according to the split in [18]: Young (18-24), Young-Adult (25-34), Middle-aged (35-49), and Senior (> 49). All private attributes are randomly split, with 10% publicly available and the remaining 90% hidden. In the experiments of data publishing, we also consider salary as the label in NBA and working field as the label in Pokec-n and Pokec-z [8, 33]. We conduct node classification on these labels as downstream tasks (using a training-testing split of 0.1:0.9), and assess the utility of published graphs by evaluating the performance of these tasks. The statistics of the three datasets are summarized in Table 6.

Implementation details. All experiments are conducted on a machine of Ubuntu 20.04 system with AMD EPYC 7763 (756GB memory) and NVIDIA RTX3090 GPU (24GB memory). All models are implemented in PyTorch version 2.0.1 with CUDA version 11.8 and Python 3.8.0. Each experiment is repeated 5 times to report the average performance with standard deviation.

For the attack model, the encoder GNN^{prox} and GNN^{role} are implemented by two 2-layer GIN [54] encoders, with the same model architecture. The hidden dimensions are set to 128 in Pokec-z, Pokec-n, and 64 in NBA. The two MLPs are both implemented by 1-layer linear transformations. The model is trained by AdamW optimizer with a learning rate of 0.001 for 300 epochs in Pokec-n, Pokec-z, and 500 epochs in NBA. For the defensive model, the sampling component consists of a two-layer Graphsage [15] encoder and a two-layer MLP. The hidden dimension of the Graphsage encoder is 64, and the hidden dimension of the MLP is 32 in all datasets. The model is trained by AdamW optimizer with a learning rate of 0.002 for 200 epochs in Pokec-n, Pokec-z, and 100 epochs in NBA. The weight decay is consistently set as 0.0005. Both models use ReLU as the non-linear activation function. For hyper-parameters settings. We perform a grid search of the degree similarity threshold (see Eq. 5) in [0,20] with a step size of 5 in all datasets. We set λ to 1, and vary γ and η (see Eq. 17) within [10,30] in NBA and (0,20] in Pokec-z and Pokec-n, with a step size of 5. Our codes are available at https://github.com/zjunet/GPS_KDD.

6.2 Experiments on Private Attribute Inference

Baselines. We compare with the following attack models, which are divided into three types: (1) MLP: multilayer perceptions; (2) GCN [26], GAT [50], GraphSAGE, Mixhop [1] and H2GCN [61]: three foundational GNNs and two heterogeneous GNNs, used as comparisons to evaluate the effectiveness of the proposed attack model in capturing privacy leakage from both proximity homophily and structure-role homophily; (3) AttriInfer [22], ComInfer [40], AI-N2V, AI-DW [13]: four methods designed for private inference on graphs. AttriInfer and ComInfer are non-deep learning models, based on Markov random fields and community detection respectively. AI-N2V and AI-DW are representation-based models that utilize Node2Vec and DeepWalk to obtain node representations.

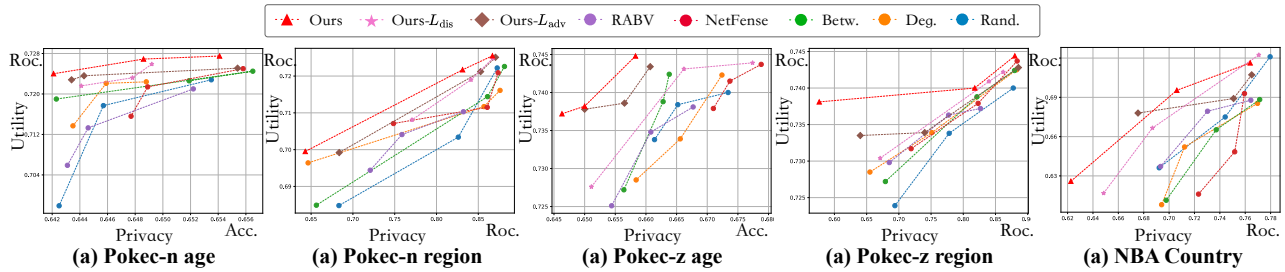
Table 1: Accuracy (age) and ROC-AUC (rest) of graph private attribute inference, where Ours denotes the proposed attack model. The best results are bolded.

	Pokec-n		Pokec-z		NBA
	Age	Region	Age	Region	Country
MLP	54.01 (3.42)	58.61 (1.36)	56.54 (3.53)	57.49 (1.10)	51.30 (3.22)
GCN	65.37 (1.03)	81.34 (0.52)	66.53 (1.28)	82.88 (0.58)	80.35 (2.21)
GAT	64.20 (1.44)	77.37 (2.28)	65.09 (2.01)	78.15 (2.74)	79.03 (4.35)
SAGE	65.29 (1.04)	81.15 (0.85)	67.14 (0.79)	82.51 (0.69)	80.71 (0.74)
MixHop	64.46 (0.50)	85.93 (0.87)	66.94 (0.38)	87.46 (0.83)	72.90 (1.89)
H2GCN	63.13 (0.27)	80.26 (1.34)	65.78 (1.61)	83.80 (1.09)	60.59 (3.31)
AttriInfer	64.83 (0.11)	63.85 (0.53)	63.18 (0.39)	63.76 (0.40)	65.93 (2.30)
ComInfer	34.18 (0.59)	62.35 (0.29)	39.64 (0.81)	55.34 (0.75)	65.97 (1.10)
AI-N2V	61.13 (0.52)	72.12 (0.42)	59.42 (0.76)	74.85 (0.87)	73.00 (1.91)
AI-DW	62.25 (0.69)	75.30 (2.18)	58.81 (0.19)	78.90 (3.61)	71.53 (2.88)
Ours	**66.69 (1.03)**	**89.39 (0.35)**	**68.80 (0.89)**	**90.01 (0.36)**	**83.32 (1.36)**

Comparison results. For evaluation, Table 1 presents the performance results of the proposed attack model in comparison with other baseline methods on the five aforementioned private attributes. The results reveal several key insights: Firstly, the notable performance drop observed in the MLP model empirically demonstrates the crucial role of GPS. Secondly, AI-N2V and AI-DW exhibit lower performance, possibly due to their limited ability to simultaneously capture both proximity and structure-role information in these representation methods. Additionally, the performance of AttriInfer and ComInfer is weaker than that of the GNN models. The reason may be attributed to the more powerful expressive capabilities of the latter models. Thirdly, the proposed attack model outperforms other baseline methods on all datasets, highlighting its remarkable efficacy in capturing GPS from both structure-role homophily and proximity homophily.

Ablation study. We conduct ablation studies to demonstrate the efficacy of each component within our attack model, including three variants: (1) Ours-prox: the inference results are solely obtained by aggregating on the entire graph; (2) Ours-role: the results are solely obtained by aggregating within each subgraph. (3) Ours-equal: dropping the GHRatios and combining the two prediction results in a 0.5:0.5 ratio. Table 2 shows the results on the five private attributes. Notably, the complete model consistently surpasses the performance of all variants, showing the effectiveness and necessity of simultaneously leveraging proximity homophily and structure-role homophily to capture privacy leakage.

6.3 Experiments on Data Publishing



(a) Pokec-n age (a) Pokec-n region (a) Pokec-z age (a) Pokec-z region (a) NBA Country

Figure 5: Privacy-utility trade-off of our defensive model and baselines. The upper-left corner represents the ideal performance.

Table 2: Ablation study of the proposed attack model.

	Pokec-n		Pokec-z		NBA
	Age	Region	Age	Region	Country
Ours-prox	65.23 (1.79)	88.00 (0.76)	67.53 (0.88)	88.96 (0.42)	82.27 (1.43)
Ours-role	65.33 (2.20)	88.37 (0.79)	66.03 (2.48)	89.01 (1.24)	79.23 (2.71)
Ours-equal	65.04 (2.31)	88.76 (0.62)	67.09 (1.62)	89.50 (0.94)	81.72 (1.65)
Ours	**66.69 (1.03)**	**89.39 (0.35)**	**68.80 (0.89)**	**90.01 (0.36)**	**83.32 (1.36)**

Baselines. We evaluate the performance of the proposed defensive model with five baselines, including: (1) Rand.: randomly dropping edges; (2) Deg./Betw. [4]: dropping edges based on the degree or betweenness centrality in descending order; (3) RABV [57]: an edge perturbation method that satisfies ϵ-edge local differential privacy, where each pair of symmetric bits in the adjacency matrix is perturbed one and only one bit; (4) NetFense [17]: a data publishing method against GNN-based inference attack on binary private attribute, the goal is to maintain data utility and protect privacy. We adopt the multi-target setting as suggested in the paper.

Comparative results. Utilizing our attack model as a worst-case adversary due to its superior performance in prior tests, we examine the privacy preservation performance by evaluating the attack models trained on the graphs generated by our defensive model and baselines. For data utility, we assess downstream task performance. Specifically, a GNN-based classifier (the same implementation as GNN^{prox} and MLP in § 6.1) is employed to conduct node classification of each dataset's label on the perturbed graphs. To ensure a fair comparison, we fix the hyper-parameters for our attack model and downstream classifier, while tuning the hyper-parameters of each defensive method to explore their privacy-utility trade-offs. We select the best three trade-off points for each method and visualize them in Figure 5. The upper-left corner of each sub-figure represents the ideal performance, with higher downstream performance and lower attack performance. Note that we also report the trade-off performance of the two variants of our defensive model, namely (1) Ours-adv: only retaining L_{adv} and L_{reg} in Eq. (17) and (2) Ours-dis: only retaining L_{dis} and L_{reg} in Eq. (17).

Figure 5 demonstrates that our defensive model consistently achieves the best privacy-utility trade-off on the five private attributes. Our variants Ours-adv and Ours-dis also demonstrate a commendable trade-off, such as in NBA country, Pokec-n age, and Pokec-z age. In contrast, methods such as Rand., Deg., and Eigen. do not take the private attribute into account during perturbation, thus compromising data utility. RABV exhibits suboptimal privacy preservation effects when introducing additional noise to the network structure. NetFense fails to adequately capture privacy leakage from both structure-role homophily and proximity homophily, thereby achieving less optimal trade-offs. In addition, it presents higher computational complexity.

Table 3: MMD distance of the degree distributions (d.d) and clustering coefficient distribution (c.c) between the original and the published graph of each method.

	Pokec-n				Pokec-z				NBA	
	Age		Region		Age		Region		Country	
	d.d	c.c	d.d	c.c	d.d	c.c	d.d	c.c	d.d	c.c
Rand.	1.354	0.897	4.184	2.847	2.057	0.777	6.546	2.539	6.442	1.356
Deg.	2.834	0.185	6.077	1.114	4.190	0.271	9.196	0.682	7.769	1.781
Betw.	2.173	0.348	5.381	1.009	3.096	0.327	8.082	1.036	7.292	1.654
RABV	2.339	1.843	7.455	3.453	3.892	1.657	8.613	2.532	8.740	2.665
NetF.	2.093	0.218	4.138	1.315	2.229	0.391	5.765	**0.596**	7.673	1.674
Ours	**0.192**	**0.118**	**1.759**	**0.945**	**0.392**	**0.073**	**3.279**	0.694	**2.498**	**0.998**

Evaluation of graph property change. To evaluate from a broader perspective, we characterize the utility by measuring the extent to which the sampled graph deviates from the original graph. Specifically, the properties of the published graph should closely resemble those of the original graph. Therefore, we employ the Maximum Mean Discrepancy (MMD) distance as our evaluation metric, comparing the degree distribution and clustering coefficient distribution of the original graph with those of the published network under our model and baselines. To ensure fairness, we tune the hyperparameters of these models to achieve comparable results in terms of privacy-preserving performance. The MMD scores in Table 3 (the smaller, the better) demonstrate that our model outperforms others on each dataset except for the Pokec-z region. These results suggest that our model, while eliminating edges associated with privacy breaches, optimally preserves the remaining graph structure, effectively maintaining data utility.

Table 4: Defending against various attack models on Pokec-n.

	AttriInfer	ComInfer	AI-N2V	AI-DW
Rand.	63.90 (0.63)	34.01 (1.01)	60.98 (0.39)	61.76 (0.77)
Deg.	63.20 (1.51)	33.62 (0.52)	60.92 (0.58)	61.23 (0.37)
Betw.	62.96 (0.63)	33.17 (0.64)	61.03 (0.89)	61.42 (0.45)
RABV	63.34 (0.81)	33.49 (0.83)	60.52 (0.45)	61.49 (0.87)
NetFense	62.84 (0.41)	32.73 (0.59)	60.01 (0.52)	61.10 (0.62)
Ours	**62.02** (**0.19**)	**31.92** (**0.63**)	**59.78** (**0.11**)	**60.86** (**0.55**)

Evaluation of transferability. As our defensive model adopts the proposed inference as the worst-case adversary during training, we aim to assess its transferability. In other words, we evaluate whether the defensive model can perform effectively against other attribute inference attack models. Table 4 reports the performance of our defensive model compared with other baselines on Pokec-n age. The results show that our defensive model outperforms other methods in protecting against various attack models, demonstrating its outstanding transferability.

Evaluation of transferability on other private attributes and more comprehensive experiments can be found in the full version.

7 RELATED WORK

Private attribute inference. Early approaches to attribute inference have primarily focused on using user-individual public attributes. These attributes include profile labels [14, 29], textual content [42, 44], and location information from users' public posts [27]. These approaches heavily rely on the correlation between public and hidden private attributes to build inference models. Despite their demonstrated effectiveness, these methods often overlook valuable information from the connections between users, resulting in a noticeable performance decline.

Subsequent explorations of attribute inference leverage network structure [13, 22, 40, 59] and involve the utilization of graph propagation algorithms, such as GCN [52] and MRF [22], to facilitate the propagation of information across connected nodes. These models aggregate information from adjacent nodes [22, 59], leverage community structures [40], or random walk [13] to infer private attributes. While they underscore the exploitation of proximity homophily [3], they often overlook the other crucial aspect of structure-role homophily, thereby achieving less than optimal performance.

Privacy-preserving learning on graph. Privacy-preserving techniques are crucial to graph data publishing, among which anonymization [11, 60], sampling-based [17], model training-based [28, 51] and differential privacy [6, 10, 31, 46, 57, 62] defense methods have been proposed. The anonymization methods [11, 60] face constraints due to the need to mitigate operational complexities and often compromise privacy and utility for efficiency. The sampling-based method [17] proposes an edge perturbation technique to defend against GNN-based inference on binary private attributes. However, it fails to consider privacy leaks from structural information and has high computational complexity. Regarding model training-based defense methods [28, 51], they often fall short when dealing with complex scenarios that require direct processing of graph data. In addition, differential privacy (DP) is a common privacy protection technique. Early efforts [6, 55, 62] extend DP to correlated settings, where data records are assumed to be correlated with each other (*e.g.*, network structure). They primarily rely on noise injection for privacy preservation. In contrast, our method systematically addresses the attack-defense problem by considering the complex relationships and structural patterns encompassed in graph data. Recently, DP-DGAE [31] perturbs the objective function of graph auto-encoders to prevent attackers from re-identifying nodes. Local DP [7] allows individuals to locally perturb their graph metrics, such as node degree and adjacency list before aggregation to mitigate the risk of privacy leakage [10, 46, 57]. Striking a balance between utility and privacy remains a challenge for them in graph data publication. Note that our method differs from DP in two key aspects: first, the determination of the edge sampling probability in DP is established according to predetermined mechanisms with respect to network structure. In contrast, our method learns the sampling probability based on the risks associated with GPS. Second, DP aims to preserve membership privacy, that is, altering one sample (*e.g.*, node or edge) doesn't significantly change the output distribution, while the privacy we investigate in this work pertains to attribute-wise privacy.

8 CONCLUSION

In this work, we delve into the problem of GPS and uncover the underlying mechanisms, including structure-role homophily, proximity homophily, and their intricate interplay. Based on this understanding, we introduce a novel data-centric approach for graph private attribute inference, capable of capturing privacy leaks from these mechanisms. Serving as the worst-case adversary, this method provides a comprehensive evaluation of potential privacy risks. To combat GPS, we propose a learnable graph sampling model for privacy-preserving data publishing. Our model enhances privacy security by learning the risks associated with each edge in GPS. Extensive experiments validate the effectiveness of our attack method and demonstrate the advantageous balance achieved by our defensive model between privacy preservation and utility retention.

ACKNOWLEDGMENTS

This work was supported in part by NSFC (62206056, 92270121, 72271059, 62322606, 62441605, 72101007), SMP-IDATA Open Youth Fund, CCF-Tencent Rhino-Bird Open Research Fund, Joint Funds of Zhejiang Provincial NSFC (LHZSD24F020001), Zhejiang Province "LingYan" Research and Development Plan Project (2024C01114), and Zhejiang Province High-Level Talents Special Support Program "Leading Talent of Technological Innovation of Ten-Thousands Talents Program" (2022R52046).

REFERENCES

[1] Sami Abu-El-Haija, Bryan Perozzi, Amol Kapoor, Nazanin Alipourfard, Kristina Lerman, Hrayr Harutyunyan, Greg Ver Steeg, and Aram Galstyan. 2019. Mixhop: Higher-order graph convolutional architectures via sparsified neighborhood mixing. In *international conference on machine learning*. PMLR, 21–29.

[2] Nesreen K Ahmed, Ryan Rossi, John Boaz Lee, Theodore L Willke, Rong Zhou, Xiangnan Kong, and Hoda Eldardiry. 2018. Learning role-based graph embeddings. *arXiv preprint arXiv:1802.02896* (2018).

[3] Faiyaz Al Zamal, Wendy Liu, and Derek Ruths. 2012. Homophily and latent attribute inference: Inferring latent attributes of twitter users from neighbors. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 6. 387–390.

[4] Aleksandar Bojchevski and Stephan Günnemann. 2019. Adversarial attacks on node embeddings via graph poisoning. In *International Conference on Machine Learning*. PMLR, 695–704.

[5] Zhipeng Cai, Zaobo He, Xin Guan, and Yingshu Li. 2016. Collective data-sanitization for preventing sensitive information inference attacks in social networks. *IEEE Transactions on Dependable and Secure Computing* 15, 4 (2016), 577–590.

[6] Rui Chen, Benjamin CM Fung, Philip S Yu, and Bipin C Desai. 2014. Correlated network data publication via differential privacy. *The VLDB Journal* 23 (2014), 653–676.

[7] Graham Cormode, Somesh Jha, Tejas Kulkarni, Ninghui Li, Divesh Srivastava, and Tianhao Wang. 2018. Privacy at scale: Local differential privacy in practice. In *Proceedings of the 2018 International Conference on Management of Data*. 1655–1658.

[8] Enyan Dai and Suhang Wang. 2021. Say no to the discrimination: Learning fair graph neural networks with limited sensitive attribute information. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 680–688.

[9] Enyan Dai and Suhang Wang. 2022. Learning fair graph neural networks with limited and private sensitive attribute information. *IEEE Transactions on Knowledge and Data Engineering* (2022).

[10] Ameya Daigavane, Gagan Madan, Aditya Sinha, Abhradeep Guha Thakurta, Gaurav Aggarwal, and Prateek Jain. 2021. Node-level differentially private graph neural networks. *arXiv preprint arXiv:2111.15521* (2021).

[11] Xiaofeng Ding, Cui Wang, Kim-Kwang Raymond Choo, and Hai Jin. 2019. A novel privacy preserving framework for large scale graph data publishing. *IEEE transactions on knowledge and data engineering* 33, 2 (2019), 331–343.

[12] Yuxiao Dong, Yang Yang, Jie Tang, Yang Yang, and Nitesh V Chawla. 2014. Inferring user demographics and social strategies in mobile social networks. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 15–24.

[13] Vasisht Duddu, Antoine Boutet, and Virat Shejwalkar. 2020. Quantifying privacy leakage in graph embedding. In *MobiQuitous 2020-17th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*. 76–85.

[14] Quan Fang, Jitao Sang, Changsheng Xu, and M Shamim Hossain. 2015. Relational user attribute inference in social media. *IEEE Transactions on Multimedia* 17, 7 (2015), 1031–1044.

[15] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).

[16] Keith Henderson, Brian Gallagher, Tina Eliassi-Rad, Hanghang Tong, Sugato Basu, Leman Akoglu, Danai Koutra, Christos Faloutsos, and Lei Li. 2012. Rolx: structural role extraction & mining in large graphs. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1231–1239.

[17] I-Chung Hsieh and Cheng-Te Li. 2021. Netfense: Adversarial defenses against privacy attacks on neural networks for graph data. *IEEE Transactions on Knowledge and Data Engineering* (2021).

[18] Jian Hu, Hua-Jun Zeng, Hua Li, Cheng Niu, and Zheng Chen. 2007. Demographic prediction based on user's browsing behavior. In *Proceedings of the 16th international conference on World Wide Web*. 151–160.

[19] Kexin Huang and Marinka Zitnik. 2020. Graph meta learning via local subgraphs. *Advances in neural information processing systems* 33 (2020), 5862–5874.

[20] Jim Isaak and Mina J Hanna. 2018. User data privacy: Facebook, Cambridge Analytica, and privacy protection. *Computer* 51, 8 (2018), 56–59.

[21] Jinyuan Jia and Neil Zhenqiang Gong. 2018. {AttriGuard}: A practical defense against attribute inference attacks via adversarial machine learning. In *27th USENIX Security Symposium (USENIX Security 18)*. 513–529.

[22] Jinyuan Jia, Binghui Wang, Le Zhang, and Neil Zhenqiang Gong. 2017. Attriinfer: Inferring user attributes in online social networks using markov random fields. In *Proceedings of the 26th International Conference on World Wide Web*. 1561–1569.

[23] Zhimeng Jiang, Xiaotian Han, Chao Fan, Zirui Liu, Xiao Huang, Na Zou, Ali Mostafavi, and Xia Hu. 2022. Topology Matters in Fair Graph Learning: a Theoretical Pilot Study. (2022).

[24] T Tony Ke and K Sudhir. 2023. Privacy Rights and data security: GDPR and personal data markets. *Management Science* 69, 8 (2023), 4389–4412.

[25] Shima Khoshraftar and Aijun An. 2024. A survey on graph representation learning methods. *ACM Transactions on Intelligent Systems and Technology* 15, 1 (2024), 1–55.

[26] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[27] Jiwei Li, Alan Ritter, and Dan Jurafsky. 2014. Inferring user preferences by probabilistic logical reasoning over social networks. *arXiv preprint arXiv:1411.2679* (2014).

[28] Kaiyang Li, Guangchun Luo, Yang Ye, Wei Li, Shihao Ji, and Zhipeng Cai. 2020. Adversarial privacy-preserving graph embedding against inference attack. *IEEE Internet of Things Journal* 8, 8 (2020), 6904–6915.

[29] Muyuan Li, Haojin Zhu, Zhaoyu Gao, Si Chen, Le Yu, Shangqian Hu, and Kui Ren. 2014. All your location are belong to us: Breaking mobile social networks for automated user location tracking. In *Proceedings of the 15th ACM international symposium on Mobile ad hoc networking and computing*. 43–52.

[30] Tiancheng Li and Ninghui Li. 2009. On the tradeoff between privacy and utility in data publishing. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. 517–526.

[31] Xiaolin Li, Li Xu, Hongyan Zhang, and Qikui Xu. 2023. Differential privacy preservation for graph auto-encoders: A novel anonymous graph publishing model. *Neurocomputing* 521 (2023), 113–125.

[32] Derek Lim, Felix Hohne, Xiuyu Li, Sijia Linda Huang, Vaishnavi Gupta, Omkar Bhalerao, and Ser Nam Lim. 2021. Large scale learning on non-homophilous graphs: New benchmarks and strong simple methods. *Advances in Neural Information Processing Systems* 34 (2021), 20887–20902.

[33] Hongyi Ling, Zhimeng Jiang, Youzhi Luo, Shuiwang Ji, and Na Zou. 2022. Learning fair graph representations via automated data augmentations. In *The Eleventh International Conference on Learning Representations*.

[34] Xiangyu Liu, Chenghao Deng, Yanchao Sun, Yongyuan Liang, and Furong Huang. 2024. Beyond Worst-case Attacks: Robust RL with Adaptive Defense via Non-dominated Policies. *arXiv preprint arXiv:2402.12673* (2024).

[35] Yang Liu, Xiang Ao, Fuli Feng, and Qing He. 2022. UD-GNN: Uncertainty-aware Debiased Training on Semi-Homophilous Graphs. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1131–1140.

[36] Chris J Maddison, Andriy Mnih, and Yee Whye Teh. 2016. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712* (2016).

[37] Abdul Majeed, Safiullah Khan, and Seong Oun Hwang. 2022. A comprehensive analysis of privacy-preserving solutions developed for online social networks. *Electronics* 11, 13 (2022), 1931.

[38] Abdul Majeed and Sungchang Lee. 2020. Anonymization techniques for privacy preserving data publishing: A comprehensive survey. *IEEE access* 9 (2020), 8512–8545.

[39] Miller McPherson, Lynn Smith-Lovin, and James M Cook. 2001. Birds of a feather: Homophily in social networks. *Annual review of sociology* 27, 1 (2001), 415–444.

[40] Alan Mislove, Bimal Viswanath, Krishna P Gummadi, and Peter Druschel. 2010. You are who you know: inferring user profiles in online social networks. In *Proceedings of the third ACM international conference on Web search and data mining*. 251–260.

[41] Nnamdi Johnson Ogbuke, Yahaya Y Yusuf, Kovvuri Dharma, and Burcu A Mercangoz. 2022. Big data supply chain analytics: ethical, privacy and security challenges posed to business, industries and society. *Production Planning & Control* 33, 2-3 (2022), 123–137.

[42] Jahna Otterbacher. 2010. Inferring gender of movie reviewers: exploiting writing style, content and metadata. In *Proceedings of the 19th ACM international conference on Information and knowledge management*. 369–378.

[43] Anurag Ranjan, Joel Janai, Andreas Geiger, and Michael J Black. 2019. Attacking optical flow. In *Proceedings of the IEEE/CVF international conference on computer vision*. 2404–2413.

[44] Delip Rao, David Yarowsky, Abhishek Shreevats, and Manaswi Gupta. 2010. Classifying latent user attributes in twitter. In *Proceedings of the 2nd international workshop on Search and mining user-generated contents*. 37–44.

[45] Leonardo FR Ribeiro, Pedro HP Saverese, and Daniel R Figueiredo. 2017. struc2vec: Learning node representations from structural identity. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. 385–394.

[46] Sina Sajadmanesh, Ali Shahin Shamsabadi, Aurélien Bellet, and Daniel Gatica-Perez. 2023. Gap: Differentially private graph neural networks with aggregation perturbation. In *USENIX Security 2023-32nd USENIX Security Symposium*.

[47] Salman Salamatian, Amy Zhang, Flavio du Pin Calmon, Sandilya Bhamidipati, Nadia Fawaz, Branislav Kveton, Pedro Oliveira, and Nina Taft. 2015. Managing your private and public data: Bringing down inference attacks against your privacy. *IEEE Journal of Selected Topics in Signal Processing* 9, 7 (2015), 1240–1255.

[48] Reza Shokri, George Theodorakopoulos, and Carmela Troncoso. 2016. Privacy games along location traces: A game-theoretic framework for optimizing location privacy. *ACM Transactions on Privacy and Security (TOPS)* 19, 4 (2016), 1–31.

Table 5: Major notations used in this work.

Notation	Definition
G, V, E	Graph, node set, and edge set
X, A	Attribute matrix and adjacency matrix
Z_L, Z_U	Known/hidden private attributes
G', E'	Sampled graph and sampled edge set
$\delta()$	Homophily indicator
$\mathcal{N}(), g()$	Neighborhood and ego network
$\text{GNN}^{\text{prox}}, \text{GNN}^{\text{role}}$	GNN encoders
$H^{\text{prox}}, H^{\text{role}}$	Node representations
$S_i, \mathcal{L}_i$	Induced subgraph and Laplacian matrix
$\Psi()$	1-hop graph filter
Θ, Φ	Model parameters
η, λ, γ	Hyper-parameters
$\hat{Z}$	Inferred private attributes
C	Number of private attribute's classes
$\mathcal{T}_{ij}, A'_{ij}$	Sampling probability, sampled adjacency
$P(), \hat{P}()$	Probability and its empirical estimation
σ	Sigmoid function
$\oplus$	Concatenation operation

[49] Susheel Suresh, Vinith Budde, Jennifer Neville, Pan Li, and Jianzhu Ma. 2021. Breaking the limit of graph neural networks by improving the assortativity of graphs with local mixing patterns. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 1541–1551.

[50] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).

[51] Binghui Wang, Jiayi Guo, Ang Li, Yiran Chen, and Hai Li. 2021. Privacy-preserving representation learning on graphs: A mutual information perspective. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 1667–1676.

[52] Le Wu, Yonghui Yang, Kun Zhang, Richang Hong, Yanjie Fu, and Meng Wang. 2020. Joint item recommendation and attribute inference: An adaptive graph convolutional network approach. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 679–688.

[53] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. 2020. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems* 32, 1 (2020), 4–24.

[54] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2018. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826* (2018).

[55] Bin Yang, Issei Sato, and Hiroshi Nakagawa. 2015. Bayesian differential privacy on correlated data. In *Proceedings of the 2015 ACM SIGMOD international conference on Management of Data*. 747–762.

[56] Liang Yang, Wenmiao Zhou, Weihang Peng, Bingxin Niu, Junhua Gu, Chuan Wang, Xiaochun Cao, and Dongxiao He. 2022. Graph neural networks beyond compromise between attribute and topology. In *Proceedings of the ACM Web Conference 2022*. 1127–1135.

[57] Qingqing Ye, Haibo Hu, Man Ho Au, Xiaofeng Meng, and Xiaokui Xiao. 2020. LF-GDPR: A framework for estimating graph metrics with local differential privacy. *IEEE Transactions on Knowledge and Data Engineering* 34, 10 (2020), 4905–4920.

[58] Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. 2021. Do transformers really perform badly for graph representation? *Advances in Neural Information Processing Systems* 34 (2021), 28877–28888.

[59] Hang Zhang, Yajun Yang, Xin Wang, Hong Gao, and Qinghua Hu. 2022. MLI: A Multi-level Inference Mechanism for User Attributes in Social Networks. *ACM Transactions on Information Systems* 41, 2 (2022), 1–30.

[60] Nannan Zhou, Shigong Long, Hai Liu, and Hai Liu. 2022. Structure–Attribute Social Network Graph Data Publishing Satisfying Differential Privacy. *Symmetry* 14, 12 (2022), 2531.

[61] Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. 2020. Beyond homophily in graph neural networks: Current limitations and effective designs. *Advances in neural information processing systems* 33 (2020), 7793–7804.

[62] Tianqing Zhu, Ping Xiong, Gang Li, and Wanlei Zhou. 2014. Correlated differential privacy: Hiding information in non-IID data set. *IEEE Transactions on Information Forensics and Security* 10, 2 (2014), 229–242.

A APPENDIX

A.1 Notations

To facilitate clarity in our presentation, Table 5 summarizes major notations used in this work.

A.2 Proof of Theorem 1

Lemma 1. *For any $A \in \mathbb{R}^{n\times d}$, let $a_i \in \mathbb{R}^{1\times d}$ denote the i-th row of A, we have*

$$\|a_i\|_2 \leq \|A\|_2, \forall 1 = 1, ..., n \tag{18}$$

Proof. Let $e_i = [0, ..., 1, ..., 0]^T$ be a vector of zero, except for the i-th element. By the definition of matrix norm, we have $\|A^T e_i\|_2 \leq \|A^T\|_2\|e_i\|_2$ and $\|A^T\|_2 = \|A\|_2$. Then we have $\|a_i\|_2 = \|A^T e_i\|_2 \leq \|A^T\|_2\|e_i\|_2 = \|A\|_2$. □

We now give the proof of Theorem 1.

Proof. Without loss of generality, we prove Theorem 1 by instantiating the encoder as a K-layer GCN encoder and a 1-hop graph filter $\Psi(\mathcal{L}) = Id - \mathcal{L}$. For simplicity, we denote the l-th layer's node representation in S_i as $H^l_{S_i}$, which is obtained as

$$H^l_{S_i} = \sigma(\Psi(\mathcal{L}_i)H^{l-1}_{S_i}W^l) \tag{19}$$

where σ denotes a τ_σ-Lipschitz activation function, $W^l \in \mathbb{R}^{d\times d}$ denotes the learnable parameters in the l-th layer.

Assume that S_i and S_j have the same number of nodes, and $\max_l \|H^l_{S_i}\|_2 \leq \tau_h$ and $\max_l \|W^l\|_2 \leq \tau_w$. Then $\forall l = 1, ..., K$, we have

$$\begin{aligned}
\|H^l_{S_i} - H^l_{S_j}\|_2 \leq& \|\sigma(\Psi(\mathcal{L}_i)H^{l-1}_{S_i}W^l) - \sigma(\Psi(\mathcal{L}_j)H^{l-1}_{S_j}W^l)\|_2 \\
\leq& \tau_\sigma\|\Psi(\mathcal{L}_i)H^{l-1}_{S_i}W^l - \Psi(\mathcal{L}_j)H^{l-1}_{S_j}W^l\|_2 \\
\leq& \tau_\sigma\tau_w\|\Psi(\mathcal{L}_i)H^{l-1}_{S_i} - \Psi(\mathcal{L}_j)H^{l-1}_{S_j}\|_2 \\
\leq& \tau_\sigma\tau_w\|\Psi(\mathcal{L}_i)H^{l-1}_{S_i} - \Psi(\mathcal{L}_j)H^{l-1}_{S_i} \\
&+ \Psi(\mathcal{L}_j)H^{l-1}_{S_i} - \Psi(\mathcal{L}_j)H^{l-1}_{S_j}\|_2 \\
\leq& \tau_\sigma\tau_w\tau_h\|\Psi(\mathcal{L}_i) - \Psi(\mathcal{L}_j)\|_2 \\
&+ \tau_\sigma\tau_w\|\Psi(\mathcal{L}_j)\|_2\|H^{l-1}_{S_i} - H^{l-1}_{S_j}\|_2
\end{aligned} \tag{20}$$

The above equation be equivalently rewritten as $R_l \leq a + bR_{l-1}$, then we have

$$\begin{aligned}
R_l &\leq a + bR_{l-1} \\
&\leq a(b+1) + b^2R_{l-2} \\
&\cdots \\
&\leq \frac{b^l - 1}{b-1}a + b^lR_0
\end{aligned} \tag{21}$$

Let $l = K$ and let $\|H_{S_i} - H_{S_j}\|_2 = \|H^K_{S_i} - H^K_{S_j}\|_2$ denote the representation difference in the last layer, we have

$$\|H_{S_i} - H_{S_j}\|_2 \leq \frac{(\tau_\sigma\tau_w)^K\|\Psi(\mathcal{L}_j)\|_2^K - 1}{\tau_\sigma\tau_w\|\Psi(\mathcal{L}_j)\|_2 - 1}\tau_\sigma\tau_w\tau_h\|\Psi(\mathcal{L}_i) - \Psi(\mathcal{L}_j)\|_2 + (\tau_\sigma\tau_w)^K\|\Psi(\mathcal{L}_j)\|_2^K\|X_i - X_j\|_2 \quad (22)$$

Since we assume that the attribute of each node is a vector of ones, we have $\|X_i - X_j\|_2 = 0$. Since the graph Laplacians are normalized, we assume $\min_l \|\Psi(\mathcal{L}_j)\|_2 \leq \tau_l$. Thus

$$\begin{aligned}\|H_{S_i} - H_{S_j}\|_2 &\leq \frac{(\tau_\sigma\tau_w\tau_l)^K - 1}{\tau_\sigma\tau_w\tau_l - 1}\tau_\sigma\tau_w\tau_h\|\Psi(\mathcal{L}_i) - \Psi(\mathcal{L}_j)\|_2 \\ &\leq \tau\|\mathcal{L}_i - \mathcal{L}_j\|_2 \end{aligned} \quad (23)$$

where $\tau = \frac{(\tau_\sigma\tau_w\tau_l)^K - 1}{\tau_\sigma\tau_w\tau_l - 1}\tau_\sigma\tau_w\tau_h$.

By a pooling function of mean, we obtain the final center node representation $H_i^{\text{role}}, H_j^{\text{role}}$ of S_i, S_j. From Lemma 1, we have:

$$\|H_i^{\text{role}} - H_j^{\text{role}}\|_2 \leq \frac{1}{n}\sum_{v=1}^{n}\|H_{S_i,v} - H_{S_j,v}\|_2 \leq \tau\|\mathcal{L}_i - \mathcal{L}_j\|_2 \quad (24)$$

which completes the proof. □

A.3 Training Algorithm and Complexity Analysis

Algorithm 1 Learnable graph sampling method

Input: Graph $G = (V, E, X)$, available private attributes Z_L, number of epochs t, update interval l, and γ, η, λ.
Output: The sampled graph $G' = (V, E', X)$.

```
Initialize GHRatio^prox and GHRatio^role for all nodes as 0.5.
for epoch e = 1, 2, ··· , t do
    Calculate the edge sampling probability by Eq. (11).
    Calculate the training loss by Eq. (13), Eq. (15) and Eq. (16).
    Update the parameters Θ of the attack model by maximizing Eq. (17).
    Update the parameters Φ of the sampling component by minimizing Eq. (17).
    if e%l == 0 then
        Update GHRatio^prox and GHRatio^role by Eq. (4) and (5).
    end if
end for
Use the learned edge sampling probability to obtain G'.
```

We divide our data publishing method into four computational steps, and we provide an analysis of the time complexity for each step.

(1) Preprocessing: In this phase, we extract the subgraphs of all nodes. Let graph $G = (V, E, X)$, the complexity of extraction is $O(n\bar{d}^k)$, where k is the number of hops, n is the number of nodes, $\bar{d}$ is the average degree of nodes. This step is computed only once.
(2) Sampling: In this phase, we compute the sampling probability for each edge and perform graph sampling. The complexity of obtaining node representations for each node through GraphSAGE is $O(rnKt)$, where r is the number of sampled neighbors, K is the number of layers, and t is the number of iterations. Here, we omit the time complexity of matrix operations. Then, obtaining edge sampling probability and performing sampling has a complexity of $O(m)$, where m represents the number of edges. Therefore, the total complexity of this step is $O(rnKt + m)$.
(3) Inference: In this phase, we perform attribute inference on the sampled graph. Firstly, the time complexity of obtaining predictions using GIN on the entire graph is $O(nKt)$. To mitigate the time overhead caused by repeated subgraph extraction, we store the global masks of corresponding edges in each node's k-hop subgraph during the preprocessing step. After each sampling step, we only need to determine which edges in the k-hop subgraph of each node are retained based on these masks. And using GIN on the subgraph for prediction has a time complexity of $O(n\bar{d}^kKt)$. Thus, the overall complexity of this step is $O(n\bar{d}^kKt)$.
(4) Loss calculation: the complexities of computing L_{adv}, L_{dis}, and L_{reg} are $O(n)$, $O((\bar{d} + \bar{b})n)$, and $O(m)$, Where $\bar{d}$ denotes the average degree (number of neighbors) for a node, and $\bar{b}$ denotes the average number of nodes with a similar degree to a given node. The overall complexity of this step is $O((\bar{d} + \bar{b})n + m)$

A.4 Datasets statistics

We provide the statistics of the three used datasets in this work.

Table 6: The statistics of datasets

Dataset	Pokec-n	Pokec-z	NBA
# nodes	66,569	67,797	403
# node attr.	59	59	39
# edges	729,129	882,765	16,570
Private attr.	region/age	region/age	country
Label	working field	working field	salary