Heterogeneous Graph Neural Network with Semantic-aware Differential Privacy Guarantees

Yuecen Wei
Guangxi Normal University
Xingcheng Fu
Beihang University
Dongqi Yan
Guangxi Normal University
Qingyun Sun
Beihang University
Hao Peng
Beihang University
Jia Wu
Macquarie University
Jinyan Wang (wangjy612@gxnu.edu.cn)
Guangxi Normal University
Xianxian Li
Guangxi Normal University

Research Article

Keywords: Heterogeneous graph, Semantic, Differential Privacy, Graph embedding

Posted Date: January 10th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-2446279/v1

Additional Declarations: No competing interests reported.

Version of Record: A version of this preprint was published at Knowledge and Information Systems on May 15th, 2023. See the published version at https://doi.org/10.1007/s10115-023-01895-6.

Heterogeneous Graph Neural Network with Semantic-aware Differential Privacy Guarantees

Yuecen Wei[1,2], Xingcheng Fu[3,4], Dongqi Yan[1,2], Qingyun Sun[3,4], Hao Peng[3], Jia Wu[5], Jinyan Wang[1,2](✉) and Xianxian Li[1,2](✉)

[1]Guangxi Key Lab of Multi-source Information Mining & Security, Guangxi Normal University, Guilin, 541004, Guangxi, China;

[2]School of Computer Science and Engineering, Guangxi Normal University, Guilin, 541004, Guangxi, China;

[3]Beijing Advanced Innovation Center for Big Data and Brain Computing, Beihang University, Beijing 100191, China;

[4]School of Computer Science and Engineering, Beihang University, Beijing 100191, China;

[5]School of Computing, Macquarie University, Sydney, Australia

Abstract. Most social networks can be modeled as a heterogeneous graph. Recently, advanced graph learning methods exploit the rich node properties and topological relationships for downstream tasks. That means that more private information is embedded in the representation. However, the existing privacy-preserving methods only focus on protecting the single type of node attributes or relationships, which neglect the significance of high-order semantic information. To address this issue, we propose a novel **Hete**rogeneous graph neural network with **S**emantic-aware **D**ifferential privacy **G**uarantees named **HeteSDG**, which provides a double privacy guarantee and performance trade-off in terms of both graph features and topology. In particular, we first reveal the privacy leakage in heterogeneous graph and define a membership inference attack with a semantic enhancement (MIS) that will improve the means of member inference attacks by obtaining side background knowledge through semantics. Then we design a two-stage mechanism, which includes the feature attention personalized mechanism and the topology gradient perturbation mechanism, where the privacy-preserving technologies are based on differential privacy. These mechanisms will defend against MIS attacks and provide stronger interpretation, but simultaneously bring in noise for representation learning. To better balance the noise perturbation and learning performance, we utilize a bi-level optimization pattern to allocate a suitable privacy budget for the above two modules. Our experiments on four public benchmarks conduct

✉*Corresponding author.*
Email:{wangjy612,lixx}@gxnu.edu.cn

performance experiments, ablation studies, inference attack verification, etc. The results show the privacy protection capability and generalization of HeteSDG.

Keywords: Heterogeneous graph; Semantic; Differential Privacy; Graph embedding

1. Introduction

Many existing works represent social networks with graphs, which make the data with multi-nodes and multi-relationships constitute heterogeneous graph [1, 2]. It makes up a more complex semantic structure and faces the fusion of multi-node features and the representation of multi-relationship structures. Representation learning for heterogeneous graph properties facilitates neural network models to learn latent information and use it for downstream tasks, such as recommendation systems [3, 4]. However, in most cases, they only exploit some explicit information such as nodes and edges, without considering higher-order and implicit semantic information.

To adapt to the heterogeneity of social networks, heterogeneous graph neural networks (HGNNs) are popular in graph representation learning because of their powerful representational capabilities [5–9]. In the existing works, to learn the dependencies of node to neighbors and high-order nodes in graphs [10–16], we divided HGNNs into two main categories: neighbor aggregation based on convolutional kernel and random walk based on meta-path, meta-graph, and meta-schema [1, 17–19]. They are achieved remarkable achievements, where the meta-path is regarded as semantic information. However, in real-world applications, powerful HGNNs boost the downstream representation ability while leading to an additional risk of privacy leakage. They mine the implicit information on the social graph, but more private information is undoubtedly implied in the representation results. For example, some malicious behaviors can deliberately push insecure links based on community detection to get private information, like social relationships [20], behavioral trajectories [21–23], and medical records. The existing works focus on how to improve the representational power of heterogeneous graph and ignore the security issues of private information.

While people benefit from the convenience of the HGNNs, they are faced with recorded behavior data and learned and used all aspects of information. That would bring a series of privacy leakage risks which are reflected in two important graph properties, i.e., features and topology. Fortunately, to address privacy problems, we can be inspired by some prominent works on common protection methods in machine learning and privacy-preserving technology for homogeneous graphs [20, 24–26]. They perturb the node features before they are input, or add noise to the gradient of the model learning. And they use some popular and advanced privacy protection technology like differential privacy [27, 28], which is based on data distribution perturbation and with a strict mathematical definition. However, existing differential privacy-based designs are difficult to adapt to semantic data and are not resistant to MIS attacks. We use the recommendation systems as an example to illustrate our MIS attack scenario. Fig. 1 (a) is a heterogeneous graph schema that expresses the node types and edge types in a social network. For feature MIS in Fig. 1 (b), it indicates that a meta-path "UserA-Item1-Store-Item2-UserC (UA-I1-S-I2-UC)" as semantic information will provide the similarity between nodes to infer the potential purchase records and

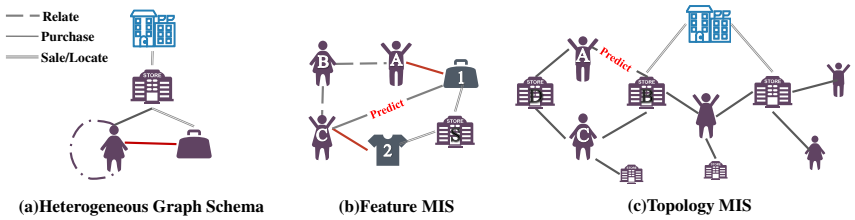


Fig. 1. A toy example of privacy risk in a heterogeneous graph.

judge whether the records are in the training set. Some traditional methods protect nodes from malicious inference attacks by perturbing homogeneous node correlations and reducing the predicted probability. However, multi-types of nodes and relationships in the heterogeneous graph will enhance the probability of a successful attack. For topology MIS, we assume that a malicious attacker can seek public networks with similar topologies and infer information about the target network by comparing topological attributes. The background knowledge allows attacker to obtain connections between arbitrary nodes regardless of node types and analyze the semantics to get the preferences of a particular user. Fig. 1 (c) reveals that a meta-graph "StoreD-UserA-StoreB-UserC (SD-UA-SB-UC)" as a semantic relationship will benefit the attacker to extract topology characteristics and infer whether the nodes are used for training models by comparing them with similar networks. In conclusion, owing to the high-order features and topological complexity caused by semantics, a unified framework needs to be designed to implement privacy protection. We propose three main challenges: (1) heterogeneous semantic information as higher-order information enhances the attacker's inference ability when node features are aggregated; (2) even if the topology is changed, heterogeneous semantic information as side information will complement the latent topological information and enhance attack inference; (3) it is difficult to trade off privacy guarantees and compelling predictions.

To resolve the above problem, we propose a novel **Hete**rogeneous Graph Neural Network with **S**emantic-aware **D**ifferential Privacy **G**uarantees method named **HeteSDG**[1]. First, we define a novel privacy leakage scenario for heterogeneous graph recommendation systems, and we reveal the privacy leakage risks associated with the heterogeneity. Further, we design two stages of privacy-preserving strategies about feature attention personalized mechanism (FeatADP) and topology gradient perturbation mechanism (TopoGDP). The FeatADP is based on a heterogeneous attention mechanism to learn and perturb the node representations. And the degree of perturbation depends on the sensitivity of features' gaussian noise [29], which is influenced by the different types of neighbors and relationships. Specifically, we follow meta-paths as semantic information to build node neighbors and make them debias for semantic attention since the build process would diminish the semantic information representation. For TopoGDP, we input the perturbed node representations to heterogeneous variational graph auto-encoders [30] for reconstructing the privacy-preserving topology, and design a regularization term as the soft supervision objective of side semantic relationships.

[1] The source code is released at `https://github.com/AixWinnie/HeteDP`.

And the link predictor can set learnable gradient clipping hyperparameters as noise sensitivity to clip and perturb the gradients. In addition, to better integrate these two properties, we use a bi-level optimization mechanism to achieve a trade-off between privacy-preserving and performance optimization and a reasonable privacy budget allocation.

A preliminary version of this work focuses on the Heterogeneous Graph Neural Network for Privacy-Preserving Recommendation (HeteDP) is published in the proceedings of ICDM 2022 [31]. This journal version has extended our method from the traditional Markovian walk into personalized awareness enhancement for semantic representation, and designed a semantic-centric regularization term. In other words, we extend the two-stage mechanism using a semantic-aware debias mechanism. In addition, we add more detailed technical explanations to the journal version and add attack experiments for more extensive datasets and analyses to evaluate our extended model HeteSDG.

The organization of this paper is listed as follows. We revisit previous works on related topics in Section 2. We introduce the problem definition in Section 3, and the overall proposed method in Section 4. The experimental results and analysis are presented in Section 5. In the last Section 6, we present the conclusion of this work.

2. Related Work

2.1. Heterogeneous Graph Embedding

Heterogeneous Graph Neural Networks (HGNNs) [32–34] are proposed to deal with ubiquitous heterogeneous data. We divide HGNNs into two main categories: neighbor aggregation based on convolutional kernel and random walk based on meta-path, meta-graph, and meta-schema. To learn high-order information, some existing models performed graph convolution directly on the original heterogeneous graph. HGT [35] proposed a transformer-based model for handling large academic heterogeneous graph with heterogeneous subgraph sampling. RGCN [33] captured the heterogeneity of graphs by projecting node embeddings into different relational spaces using multi-relational aggregation weight matrices. HetSANN [36] used a type-specific graph attention layer for the aggregation of local information, avoiding manually selecting meta-paths.

In addition to mining explicit node features and topology structure in graphs, some works extracted semantic information as additional guidance for heterogeneous graph embedding by adding meta-paths, meta-graph, or meta-schema as prior knowledge to effectively fuse heterogeneous data and improve learning performance. HetGNN [32] considered each node's heterogeneous content (node's attribute information) and used the random walk to sample a fixed number of strongly associated heterogeneous neighbors for graph nodes, and then used BiLSTMs to process the heterogeneous graph. Metapath2vec [18] designed a meta-path based random walk and utilized skip-gram to generate node embeddings. HIN2Vec [17] learned node representations based on meta-path random walks to incorporate semantic information in heterogeneous graph. HGConv [35] introduced node representation based on mixed micro/macro level convolution operations on heterogeneous graph. A micro-level convolution can learn the dependency of nodes under the constraints of the same relation, and a macro-level convolution was used to distinguish subtle differences between relation types.

With the contextualization of advanced research, many works had also made excellent progress in recommendations [37, 38]. However, with the representation and application of rich information, more user information is undoubtedly exposed, and the adversary collects the side information of the node features, topology structure, or semantic information from public sources to infer the private information.

2.2. Graph Privacy Protection

Since graph neural networks (GNNs) play a crucial role in deep learning, the privacy problems in graph embedding are exposed, and some early work tried to protect graph data privacy and achieve meaningful results.

For homogeneous graphs, some works preserved the user information by personalized privacy protection [39] and the use of anonymization mechanisms [40, 41] prevented an attacker from inferring private information. Recently, DPGGAN [28] had carried out differential privacy in GNNs by referring to DP-SDG [42] privacy-preserving design patterns and taking advantage of VGAE [30]. LDP [43] disturbed local user features to protect the privacy of node features and solved precision degradation by excessive injection noise through KProp. PrivGnn [44] randomly sampled private data from the training set and input the sampled data into the model for training pseudo labels, and then mixed pseudo labels in the public data to achieve privacy protection. In application, GERAI [27] was a recommendation model, in which a combination of GNNs and LDP ensures the practicability of learning and protects users' information from attribute inference attacks.

For heterogeneous graph, there was new work on the design of a heterogeneous differential privacy mechanism [45] whose target was to solve the problem of privacy budget allocation due to the different distribution of heterogeneous data. However, with the addition of more side information (semantic information), the inference capability of the attackers may be enhanced, and the existing methods are difficult to adapt to the high-order feature and topological complexity caused by semantics.

3. Preliminaries and Problem Definition

To efficiently implement privacy protection for heterogeneous graph, we use differential privacy techniques [46] (DP) that are consistent with our data type and framework design. Differential privacy is recognized as one of the quantifiable and practical privacy-preserving models. The basic idea is that any computation cannot be significantly affected by any operation such as add, delete and modify. Even if the attackers know all records except this one, they cannot obtain any information from it.

(ϵ, δ)-Differential Privacy [47]. A random algorithm $\mathcal{M}$ satisfies (ϵ, δ)-Differential Privacy for any two neighboring data sets D and D' and any possible subset of output $\mathcal{O} \subseteq Range(\mathcal{M})$, and it holds that

$$\Pr[\mathcal{M}(D) \in \mathcal{O}] \leq e^{\epsilon} \Pr[\mathcal{M}(D') \in \mathcal{O}] + \delta. \tag{1}$$

where the nodes D and D' differ by at most one record. The privacy strength of DP increases as the privacy budget decreases, which is controlled by ϵ and δ.

Table 1. Summary of notations.

Symbol	Description
G	Heterogenous graph
ϵ	Global privacy budget
ϵ_f	Feature perturbation privacy budget
ϵ_s	Topology perturbation privacy budget
$\mathcal{M}$	Gaussian mechanism
$\triangle_2\mathcal{S}$	Sensitivity
$\mathcal{N}_{feat}$	Feature noise-adding function
$\mathcal{N}_{topo}$	Topology noise-adding function
m	Meta-path
$\mathcal{C}$	Semantic-aware debias weight
$\mathcal{A}_u^m$	Neighbor attention coefficient of node u on m
$\mathcal{B}_m$	Semantic Attention coefficient on m
C	Gradient norm bound
$\widetilde{\mathrm{h}}$	Perturbed features
$\widetilde{\mathrm{g}}$	Perturbed gradient
$\mathbf{X}, \mathbf{h}$	Node representation
$\mathbf{A}$	Relationship matrix
$q(\cdot)$	Feature encoder
$p(\cdot)$	Link predictor

Thus, (ϵ, δ)-DP is guaranteed by adding appropriate noise to the output of the algorithm, and the amount of injected noise is calibrated to the sensitivity.

Sensitivity [47]. Given any query $\mathcal{S}$ on D, the sensitivity for any neighboring data sets D and D' which is defined as

$$\Delta_2\mathcal{S} = \max_{D,D'} \|\mathcal{S}(D) - \mathcal{S}(D')\|_2 . \tag{2}$$

Gaussian Mechanism [29]. Let $\mathcal{S} : D \to \mathbb{O}^{\mathcal{K}}$ be an arbitrary $\mathcal{K}$-dimensional function and define its l_2 sensitivity to be $\Delta_2\mathcal{S}$. The Gaussian Mechanism with parameter σ adds noise scaled to $\mathcal{N}\left(0, \sigma^2\right)$ to each of the $\mathcal{K}$ components of the output. Given $\epsilon \in (0, 1)$, the Gaussian Mechanism is (ϵ, δ)-DP with

$$\sigma \geq \sqrt{2\ln(1.25/\delta)}\Delta_2\mathcal{S}/\epsilon. \tag{3}$$

Adding noise is the primary means to implement privacy-preserving by differential privacy. In this work, we will apply Gaussian noise to the node features and link prediction gradients of the heterogeneous graph G, respectively, and the overall form is defined as

$$\mathcal{M}(G) \triangleq \mathcal{S}(G) + \mathcal{N}\left(0, (\triangle_2\mathcal{S})^2 \sigma^2\right), \tag{4}$$

where $\Delta_2\mathcal{S}$ controls the amount of noise in the generated Gaussian distribution from which we will sample noise into the target.

Privacy Leakage Analysis. Some existing works [28] only consider the high-order influence of the same node types while reducing the probability of malicious attackers stealing user interest orientations by perturbing their edges. However, the heterogeneous graph with complex data in which will enhance the attackers' inference ability i.e. the attacker can obtain the membership of nodes by inferring the semantic information from other types of nodes. Consequently, the existing privacy-preserving methods are hard to adapt to the high-order features and topological complexity caused by semantics. And MIS in HGNNs will utilize node features and topology structure to further formulate semantic links to infer private information. Therefore, we transform the privacy problem

on heterogeneous graph into an associative differential privacy problem of graphs with solid semantic correlation. This means that our problem further becomes a multi-objective optimization problem for representation learning as well as optimal privacy budget allocation. The key symbols of this paper are summarized in Table 1.

Problem Definition. We aim to maximize the privacy-preserving effect while minimizing the information loss due to the noise. Therefore, we combine optimal privacy budget allocation with model optimization as a multi-objective optimization problem. There is a heterogeneous graph $G = (V, E, \phi, \psi)$ with an entity mapping function $\phi(v) : V \to A$ and a relation mapping function $\psi(e) : E \to R$, where V and E are the set of nodes and edges. Each node $v \in V$ belongs to the node typeset A, and each edge $e \in E$ belongs to the edge typeset R. The graph has the meta-paths $m = a_1 \overset{r_1}{\to} a_2 \overset{r_2}{\to} \ldots \overset{r_{N-1}}{\to} a_N$ constructed by node type $a_i \in A\,(i = 1, 2, \ldots, N)$ and edge type $r_i \in R\,(i = 1, 2, \ldots, N)$, where $a_i = \phi(v_i)$ and $r_i = \psi(e_i) = \psi(\langle v_i, v_{i+1}\rangle)$. Then, given an objective function with FeatADP $f(x, y)$ and TopoGDP $g(x, y)$ on the privacy budget of ϵ_f and ϵ_s, the problem can be defined as follows

$$\min_{x \in \epsilon_s} g(x, y)\,,\ \mathbf{s.t.}\ y \in \arg\min_{y \in \epsilon_f} f(x, y)\,, \tag{5}$$

where global privacy budget $\epsilon = \epsilon_f + \epsilon_s$. Such problems usually difficult to find a unified optimal solution, which is the same as the multi-objective optimization in existing graph learning. We are inspired by the multi-head attention mechanism [48] and differentially private stochastic gradient descent [42]. We formulate two protection strategies for node and topology, respectively. In the next section, we will specify our proposed privacy-preserving approach.

4. Proposed Methodology

In this section, we introduce the HeteSDG framework, a heterogeneous graph neural network with semantic-aware differential privacy guarantees, and illustrate the details of privacy mechanisms and learning optimization mechanisms. Fig. 2 presents our proposed framework with the two-stage DP mechanisms of node features and topology structure. Specially, we elaborate on the HeteSDG approach (see Algorithm 1) and intuitive illustration that FeatADP will provide the perturbed features for TopoGDP to execute downstream.

4.1. Feature Attention Personalized mechanism

In this section, we detail the node-level privacy-preserving mechanism and incorporate it into feature representation learning. Specially, we compute the representation of various nodes by learning the influence weights of neighbors and semantics.

The semantic neighbor building follows a Markov chain and meta-paths, i.e. the conditional probability of the node u with type a_{i+1} at moment is determined by the node v with type a_i only, and the next node type to walk is fixed. In particular, to adapt the heterogeneity of the data, we constrain the generation of semantic neighbors through meta-paths m. In addition, we obtain natural semantic neighbor information weights $\mathcal{C}$ for preparing further conditioning semantic-level

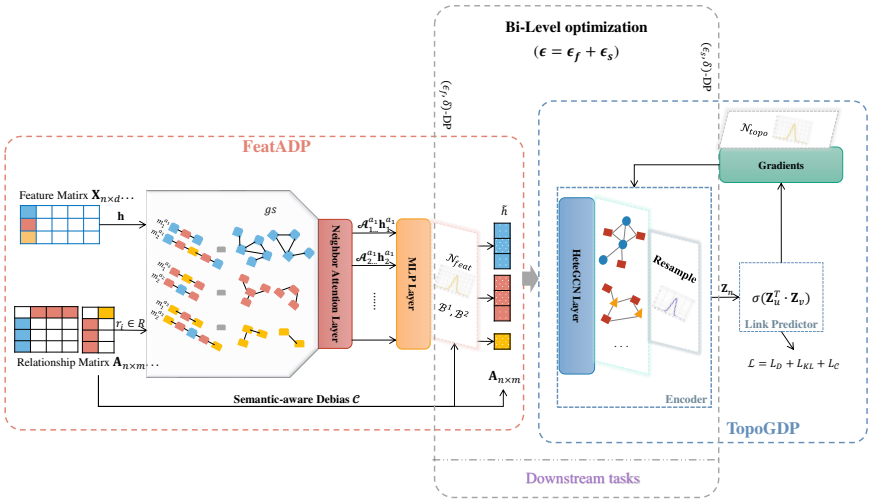


Fig. 2. The framework of HeteSDG. HeteSDG consists of two major components: FeatADP and TopoGDP. The first part secures the node attributes, and the second part protects the topology. The two parts are constrained by a global privacy budget so that the perturbation to the model is within a reasonable range and the optimal accuracy is pursued.

feature learning, which we call the **semantic-aware debias** mechanism. The formulation is expressed formally as

$$(gs_m, \mathcal{C}) \leftarrow P(a_{i+1} \,|a_i, \ldots, a_1) = P(a_{i+1} \,|a_i) = \mathbf{A}^m_{a_{i+1}a_i} = \mathbf{A}^m_{r_i}, \tag{6}$$

where $a_{i+1} = a_1$. For multi-nodes, we map them to a uniform space through linear transformation, and the embedding of the l-th layer neural network as

$$\mathbf{h}_u^{(l)} = w^{(l)} \odot \mathbf{f}_u^{(l)}, \tag{7}$$

where $\mathbf{h}_u^{(l)}$ and $\mathbf{f}_u^{(l)}$ are the embedding and the original feature of the node u, and $w^{(l)}$ is linear mapping matrices.

For neighbor-level aggregation, to learn the dependence between node u and neighbor v, we leverage the attention mechanism and normalize the overall attention value to quantify that we calculate the attention score between nodes as

$$W^{(l)}_{(uv;m)} = \frac{\exp(\sigma(\alpha(\mathbf{h}_u \parallel \mathbf{h}_v))^{(l)}_m)}{\sum_{k\in V_{gs}(u)} \exp(\sigma(\alpha(\mathbf{h}_u \parallel \mathbf{h}_k))^{(l)}_m)}, \tag{8}$$

where $V_{gs}(u)$ is the set of neighbors which includes node v and following Eq. (6), $\sigma(\cdot)$ is an activation function, α is a learnable weight vector and $\parallel$ denotes concatenation. Then, we introduce multi-head attention for node representation learning to pay attention to more comprehensive neighbor information. And we explicitly obtain the learnable influence weight of node u by other nodes simultaneously as the guiding of neighbor-level perturbation. So we design the multi-head attention coefficients and node representations between nodes on the

$(l+1)$-th layer of each subgraph as

$$\mathbf{h}_u^{(m,l+1)} = ||_{k=1}^{K} \sigma \left(\sum_{v \in V(u)} W_{(uv;m)}^{k} \mathbf{z}_v^{(l)} \right), \tag{9}$$

$$\mathcal{A}_u^{(m,l+1)} = \sigma \left(\frac{1}{K} \sum_{k=1}^{K} \sum_{v \in V(u)} W_{(uv;m)}^{k} \right), \tag{10}$$

where K is the head of multi-head attention and $\mathbf{z}_v^{(l)}$ is the embedding of the neighbor node. Specially, We use two types of multi-headed attention aggregation for node embeddings and attention weights, this design is just to better fit our data format, actually, they can be mixed.

For semantic-level aggregation, we utilize residual concatenate for the node embeddings to retain more semantic dependency as

$$\mathbf{z}_u^m = ||_{m=1}^{M} \mathbf{h}_u^m. \tag{11}$$

where M is the number of meta-paths. As we analyzed, heterogeneous data are more vulnerable to semantic inference attacks, and we further consider the impact of semantic-level on node representation. The semantic attention from multilayer perceptron (MLP) as

$$\mathcal{W}_m = \frac{1}{|V|} \sum_{u \in N} \textsc{LeakyReLU} \left(w_2 \mathbf{z}_u^m + b \right), \tag{12}$$

$$\mathcal{B}_m = \frac{\exp(\mathcal{W}_m + \mathcal{C})}{\sum_{m \in M} \exp(\mathcal{W}_m + \mathcal{C})}, \tag{13}$$

where $\mathcal{W}_m$ is the attention weight of m and $\mathcal{B}_m$ is the normalized attention coefficient. We note that $\mathcal{C}$ can further metric the awareness of semantics and provide more accurate semantic preferences following Eq. (6). So we get the multi-level embeddings as

$$\mathbf{z}_u = \sum_{m}^{M} \mathcal{B}_m \mathbf{z}_u^m. \tag{14}$$

For the privacy-preserving feature learning, we inject personalized noise into the nodes individually, which means our noise fuse the weights of neighbor and semantics. We design the sensitivity and Gaussian noise on heterogeneous graph following Eq. (2), (10) and (13) as

$$\begin{aligned} &\triangle_2 \mathcal{S}_{feat} = \max_{D,D'} \mathcal{A}_u^m \mathcal{B}_m \cdot \left\| \mathcal{S}(D) - \mathcal{S}(D') \right\|_2, \\ &\widetilde{\mathbf{h}} = \mathbf{z}_u + \lambda \cdot \mathcal{N}_{feat}^u \left(0, \sigma_{\epsilon_f}^2 (\triangle_2 \mathcal{S}_{feat})^2 \mathbf{I} \right), \end{aligned} \tag{15}$$

where λ is a hyperparameter, the privacy budget $\epsilon_f < \epsilon$ and $\mathcal{N}_{feat}^u$ is the Gaussian distribution with mean 0 and standard deviation $\sigma_{\epsilon_f} \triangle_2 \mathcal{S}_{feat}$ for u to satisfy (ϵ_f, δ)-DP.

Algorithm 1: HeteSDG.

Input: Meta-path m; Relationship matrix $\mathbf{A}$; Node set V; Node feature $\mathbf{h}$; Original sensitivity $\triangle_2\mathcal{S}$; Local privacy budget ϵ_f or ϵ_s; Number of negative sampling k; Number of training epochs T; Batch size B; Noise scale σ; Gradient norm bound C .

Output: Predicted result of the downstream task.

Initialize model parameters;

// FeatADP

$(gs_m, \mathcal{C}) \leftarrow \mathbf{A}^m$;

$W_{(uv\in V;m)} = \frac{\exp(\sigma(\alpha(\mathbf{h}_u \| \mathbf{h}_v))_m)}{\sum_{k\in V_{gs}(u)} \exp(\sigma(\alpha(\mathbf{h}_u \parallel \mathbf{h}_k))_m)}$;

Get $\mathcal{A}_u^{(m,l+1)}$ with multi-head attention from aggregating W;

Get $\mathcal{W}_m$ from MLP;

$\mathcal{B}_m = \frac{\exp(\mathcal{W}_m + \mathcal{C})}{\sum_{m\in M} \exp(\mathcal{W}_m + \mathcal{C})}$;

Update sensitivity $\triangle_2\mathcal{S}_{feat} = \mathcal{A}_u^m \mathcal{B}_m \cdot \triangle_2\mathcal{S}$;

Calculate node embeddings $\mathbf{z}_u \leftarrow (W, \mathcal{B}_m)$;

$\widetilde{\mathbf{h}} = \mathbf{z}_u + \lambda \cdot \mathcal{N}_{feat}^u \left(0, \sigma_{\epsilon_f}^2 (\triangle_2\mathcal{S}_{feat})^2 \mathbf{I}\right)$

// TopoGDP

$\widetilde{\mathbf{h}}_{v'}^k \leftarrow \textsc{NegSample}\left(\widetilde{\mathbf{h}}_u, k | \forall u \in V\right)$;

for $t = 1, 2, \ldots, T$ **do**

  Encode $\widetilde{\mathbf{h}}$ to $\mathbf{z}$ from encoder $q(\cdot)$;

  Predict $\widetilde{\mathbf{A}}$ from $p(\cdot)$

  $\mathcal{L} = \mathcal{L}_D^r + \gamma\mathcal{L}_{KL} + \eta\mathcal{L}_{\mathcal{C}}$

  Get gradient $\mathbf{g}$;

  Update sensitivity $\triangle_2\mathcal{S}_{topo} = C$;

  $\widetilde{\mathbf{g}} = \frac{1}{|B|} \left(\sum_{i\in B} \mathbf{g}_i^r / \max\left(1, \frac{\|\mathbf{g}_i^r\|_2}{\triangle_2\mathcal{S}_{topo}}\right) + \mathcal{N}_{topo}\left(0, \sigma_{\epsilon_s}^2 (\triangle_2\mathcal{S}_{topo})^2 \mathbf{I}\right)\right)$;

end

4.2. Topology Gradient perturbation mechanism

In this section, our design is based on a heterogeneous variable auto-encoder which contains an embedding encoder and link predictor. It executes the heterogeneous differential privacy stochastic gradient descent to achieve privacy protection for topology structures.

For embedding encoder. We build a two-layer heterogeneous graph neural network (HeteGCN) inspired by some state-of-art model [30, 33, 49]. And its aggregated representations of multi-nodes and multi-relationships as

$$\begin{aligned} &\mathbf{z}_{dst}^{(l+1)} = \textsc{Agg}\left(f_r\left(G, \widetilde{\mathbf{h}}_{src}^{(l)}, \widetilde{\mathbf{h}}_{dst}^{(l)}\right) |_{r\in R}\right) \\ &\textbf{s.t. } \mathbf{z} = \textsc{HeteGCN}(\mathbf{X}, \mathbf{A}_r), \end{aligned} \tag{16}$$

where f_r is the HeteGCN module of each $r \in R$, $\mathbf{X} = \widetilde{\mathbf{h}}$ is node features, and $\mathbf{A}_r$ is the relationship matrix. The hidden layer representation of each node under

the relational subgraph as

$$\mathbf{z}_u^{(l+1)} = \sigma\left(\sum_{v \in V_{(u)}} \zeta w^{(l)} \widetilde{\mathbf{h}}_v^{(l)}\right), \tag{17}$$

where ζ is a normalization constant, and $w^{(l)}$ and $\widetilde{\mathbf{h}}_v^{(l)}$ are the learnable weight matrices and neighbor node embeddings of the l-th layer, respectively.

Our training process is an unsupervised representation learning and we design negative sampling to enhance the generalizability of the model, which will compute the difference in scores between two connected nodes and any pair of nodes. For example, there is an edge $e \in E$ between nodes $u \in V$ and $v \in V$ in graph G, and we want the score between u and v to be higher than the score between u and k nodes v' sampled from an arbitrary distribution $v' \sim \mathrm{Pn}(v)$. We random extract a batch negative sample in each iteration training through the neighbor sampling of the multi-layer GNN, and the negative sampling definition as

$$\widetilde{\mathbf{h}}_{v'}^{k} \leftarrow \textsc{NegSample}\left(\widetilde{\mathbf{h}}_u, k | \forall u \in V\right). \tag{18}$$

Then, we adopt a two-layer HeteGCN model following Eq. (16) as an encoder and utilize the reparameterization trick in training

$$q\left(\mathbf{Z}|\mathbf{X}, \mathbf{A}_r\right) = \prod_{i=1}^{V} \mathrm{Pn}\left(\mathbf{z}_i | \mu_i^r, \left(\sigma_i^2\right)^r\right), \tag{19}$$

where $\mathbf{z}$ is a stochastic latent sampling variable, $\mu_r = \textsc{HeteGCN}_{\mu}\left(\mathbf{X}, \mathbf{A}_r\right)$ is the matrix of mean vectors μ_i^r and $\log_{\sigma}^r = \textsc{HeteGCN}_{\sigma}\left(\mathbf{X}, \mathbf{A}_r\right)$ is the matrix of standard deviation vectors σ_i^r.

For link predictor, we compute the inner product between latent variables to reconstruct the edge and leverage the calculation to express the connection probability of two different types of nodes $\phi(\mathbf{z}_u)$ and $\phi(\mathbf{z}_v)$ as

$$p\left(\widetilde{\mathbf{A}}_r | \mathbf{Z}\right) = \prod_{i=1}^{|A_u|} \prod_{j=1}^{|A_v|} \sigma\left(\mathbf{z}_u^T \mathbf{z}_v\right), \tag{20}$$

where $\mathbf{z}_u^T$ represents the transpose of $\mathbf{z}_u$.

The topology representation learning is to study a suitable and superior data distribution and discover latent structure. Therefore, We can learn the interdependence and association of node u and v based on semantic association rules and calculate the score between the node pair with the unsupervised cross-entropy loss of the graph as

$$\mathcal{L}_D^r = -\log \sigma\left(q_{u,v}\right) - k \cdot \mathbb{E}_{v' \sim \mathrm{Pn}(v)} \log\left(\sigma\left(-p_{u,v'}\right)\right), \tag{21}$$

where k is the number of negative sampling. To alleviate the topology perturbation, we set the KL divergence $\mathcal{L}_{KL}$ to constrain the distribution between generated samples and real samples. Specially, we design a regularization term $\mathcal{L}_{\mathcal{C}}$ as the soft supervision objective of side semantic relationship. The loss function as

$$\begin{aligned} \mathcal{L} &= \mathcal{L}_D^r + \gamma \mathcal{L}_{KL} + \eta \mathcal{L}_{\mathcal{C}} \\ &= \mathcal{L}_D^r + \gamma \sum_{i \in \langle u,v \rangle} \mathrm{KL}\left(q_i || p\left(\mathbf{Z}_i\right)\right) + \eta \frac{1}{2N} \sum_{i=1}^{V} \left\| \left(\textstyle\sum_{k \in V(u_i)} \mathbf{z}_k\right)_i^{\mathrm{T}} - \left(\mathbf{z}_i^{\mathrm{T}}\right) \right\|_2^2, \end{aligned} \tag{22}$$

where γ and η are hyperparameters, $p\left(\mathbf{Z}_i\right) = \prod_i \mathrm{Pn}\left(\mathbf{z}_i|0,\mathbf{I}\right)$ is a Gaussian prior, and $\sum_{k\in V(u_i)} \mathbf{z}_k$ is the predicted average of neighbors $V(u_i)$ to nodes u_i.

For the topology privacy-preserving learning, we protect topology by perturbing the gradient of the representation learning. And we inject the Gaussian noise to the training gradient, so the sensitivity further defines as $\triangle_2\mathcal{S}_{topo} = C$ following Eq. (4).

Then, for each iteration in training, we calculate the gradient of predictor $\mathbf{g} = \nabla\mathcal{L}$ from backpropagation, inject noise into the gradient after gradient clipping and before gradient update, and finally perform gradient descent. Thus, the perturbed gradients as

$$\widetilde{\mathbf{g}} = \frac{1}{|B|}\left(\sum_{i\in B} \mathbf{g}_i^r / \max\left(1, \frac{\|\mathbf{g}_i^r\|_2}{\triangle_2\mathcal{S}_{topo}}\right) + \mathcal{N}_{topo}\left(0, \sigma_{\epsilon_s}^2(\triangle_2\mathcal{S}_{topo})^2\mathbf{I}\right)\right), \tag{23}$$

where B and $|B|$ are the batch and size for each training iteration, respectively, $\|\mathbf{g}_i^r\|_2$ is the l_2 norm of gradient clipping, and $\mathcal{N}_{topo}(\cdot)$ is the Gaussian distribution with mean 0 and standard deviation $\sigma_{\epsilon_s}\triangle_2\mathcal{S}_{topo}$. The distribution satisfies (ϵ_s, δ)-DP, where the privacy budget $\epsilon_s < \epsilon$. We control the sensitivity to noise by limiting the norm bound C of a gradient. To adapt to the noise distribution in heterogeneous data, we utilize privacy accounting [42] to regulate the privacy budget of each iteration. We set a constant number c_2, the sampling probability P, and the number of iterations T for training to make $\sigma\epsilon_s \geq c_2 P\sqrt{T\log 1/\delta}$.

4.3. Bi-level optimization of HeteSDG

In this section, we introduce a bi-level optimization mechanism [50] to achieve a two-stage privacy budget allocation with Eq. (5). The aim is to maximize the privacy-preserving effect while minimizing the information loss due to noisy inputs. The optimization is organized into two processes. For FeatADP optimization, we fix the hyperparameter of privacy budget ϵ_s and find the optimal value of ϵ_f where $\{\epsilon_f \in \mathbb{R} : 0 < \epsilon_f < \epsilon\}$, and the approximate solution formula is

$$(\epsilon_f)_t = y_{t-1} - \nabla f(y_{t-1}, x), \tag{24}$$

where $y \in \epsilon_f$, $x \in \epsilon_s$, $t = 1, 2, \ldots T$ and ∇ denotes gradient descent. For TopoGDP optimization, we fix the hyperparameter of privacy budget ϵ_f and find the optimal value of ϵ_s where $\{\epsilon_s \in \mathbb{R} : 0 < \epsilon_s < \epsilon\}$, and the parameter update as

$$\epsilon_s = x - \nabla_x g(y, T, x). \tag{25}$$

4.4. Privacy-Preserving Analysis of HeteSDG

In this section, we give a detailed privacy analysis and proof for HeteSDG in the following theorem.

Theorem 1. A random function $\mathcal{M}$ is (ϵ, δ)-DP if the privacy loss $\mathcal{C}_{\mathcal{M}}(o, D, D')$ satisfies $\Pr\left[\mathcal{C}_{\mathcal{M}} \geq \epsilon\right] \leq \delta$, where the privacy loss define as

$$\mathcal{C}_{\mathcal{M}}(o, D, D') := \ln \frac{\Pr\left[\mathcal{M}(D) = o\right]}{\Pr\left[\mathcal{M}(D') = o\right]}.$$

Proof. Let us partition $\mathbb{O}$ as $\mathbb{O} = \mathcal{O} \cup \mathcal{O}'$, where $\mathcal{O} = \{o \in \mathbb{O} : \mathcal{C}_{\mathcal{M}} \geq \epsilon_{f,s}\}$ and $\mathcal{O}' = \{o \in \mathbb{O} : \mathcal{C}_{\mathcal{M}} < \epsilon_{f,s}\}$. For any $S \subseteq \mathbb{O}$, if $\Pr\left[\mathcal{C}_{\mathcal{M}}\left(o, D, D'\right) \geq \epsilon_{f,s}\right] \leq \delta$, we have

$$\begin{aligned}
&\Pr\left[\mathcal{M}\left(D\right) \in S\right] \\
&= \Pr\left[\mathcal{M}\left(D\right) \in S \cap \mathcal{O}\right] + \Pr\left[\mathcal{M}\left(D\right) \in S \cap \mathcal{O}'\right] \\
&\leq \Pr\left[\mathcal{M}\left(D\right) \in \mathcal{O}\right] + \exp\left(\epsilon_{f,s}\right) \Pr\left[\mathcal{M}\left(D'\right) \in S \cap \mathcal{O}'\right] \\
&\leq \delta + \exp\left(\epsilon_{f,s}\right) \Pr\left[\mathcal{M}\left(D'\right) \in S\right],
\end{aligned}$$

yielding (ϵ, δ)-DP for the Gaussian mechanism, where $\epsilon_{f,s}$ denotes the privacy budget of noise on node features or topology. □

Theorem 2. Let $\mathcal{M}_1 : D \to \mathcal{O}_1$ be an (ϵ_f, δ)-DP algorithm, and $\mathcal{M}_2 : D \to \mathcal{O}_2$ be an (ϵ_s, δ)-DP algorithm. Their combination defined to be $\mathcal{A} = \mathcal{M}_{1,2} : D \to \mathcal{O}_1 \times \mathcal{O}_2$ by the mapping: $\mathcal{A}\left(x\right) = \left(\mathcal{M}_1\left(x\right), \mathcal{M}_2\left(x\right)\right)$ is $(\epsilon_f + \epsilon_s, \delta)$-DP.

Proof. Let $x, y \in D$ and fix $\forall\left(o_1, o_2\right) \in \mathcal{O}_1 \times \mathcal{O}_2$. Then

$$\begin{aligned}
&\Pr\left[\mathcal{A}\left(D\right) = \mathcal{O}\right] + \delta \\
&= \frac{\left(\Pr\left[\mathcal{M}_1\left(x\right) = o_1\right] + \delta\right)\left(\Pr\left[\mathcal{M}_2\left(x\right) = o_2\right] + \delta\right)}{\left(\Pr\left[\mathcal{M}_1\left(y\right) = o_1\right] + \delta\right)\left(\Pr\left[\mathcal{M}_2\left(y\right) = o_2\right] + \delta\right)} \\
&= \left(\frac{\Pr\left[\mathcal{M}_1\left(x\right) = o_1\right] + \delta}{\Pr\left[\mathcal{M}_1\left(y\right) = o_1\right] + \delta}\right)\left(\frac{\Pr\left[\mathcal{M}_2\left(x\right) = o_2\right] + \delta}{\Pr\left[\mathcal{M}_2\left(y\right) = o_2\right] + \delta}\right) \\
&\leq \exp\left(\epsilon_f\right)\exp\left(\epsilon_s\right) = \exp\left(\epsilon_f + \epsilon_s\right),
\end{aligned}$$

which shows that the combination algorithm $\mathcal{A}$ satisfies $(\epsilon_f + \epsilon_s, \delta)$-DP. □

5. Experiments

5.1. Experimental Setup

In this section, we conduct experiments on four datasets and two tasks to demonstrate the adaptability of heterogeneity privacy protection and the effectiveness of heterogeneous graph learning. The experiment results of HeteSDG are shown in Table 3, where the best accuracy is shown in bold and the best privacy-preserving results are underlined. Furthermore, "−" indicates that the current model hardly implements in the dataset. We then further analyze how HeteDP and HeteSDG are affected by changing the strength of privacy-preserving, and our contribution to the overall performance of the optimization model.

Datasets. We use four open social network datasets, including citation networks (ACM and DBLP), an E-commerce dataset (Amazon), and a relational movie network (IMDB). The dataset statistics are shown in Table 2 where we mark the classified nodes and the predicted edges with bolded. For example, in the downstream task of the ACM dataset, we perform node classification for "paper" and link prediction for "paper-author". The choices follow the majority of heterogeneous graph learning.

Baselines. We compare the HeteDP and HeteSDG with state-of-the-art heterogeneous baseline methods in different categories: (1) meta-paths-based GNNs, we select metapath2vec [18] and HetGNN [32], where HetGNN has ignored

Table 2. Statistics of Datasets.

Dataset	# Nodes(Label)	# Edges
ACM	author: 17,351 field: 72 **paper**: 4,025(3)	**paper-author**: 13,407 paper-field: 4,025
DBLP	**author**: 4,025(4) conf: 20 paper: 14,328 term: 7,723	**paper-author**: 19,645 paper-conf: 14,328 paper-term: 85,810
Amazon	category: 22 **item**: 2,753(3) user: 6,170 view: 3,857	item-category: 5,508 **item-user**: 195,791 item-view: 5,694
IMDB	actor: 5,257 director: 2,081 **movie**: 4,278(3)	**movie-actor**: 12,828 movie-director: 4,278

Table 3. Summary of experimental results: "F1 score in NC and ROC-AUC score in LP" (%).

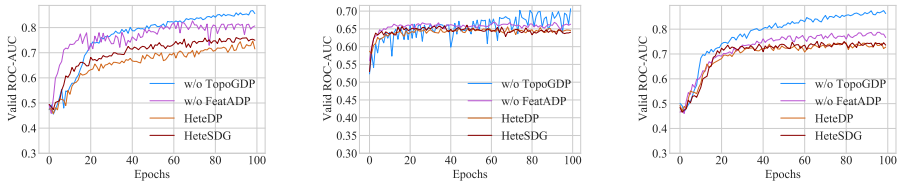
Dataset	**ACM**		**DBLP**		**Amazon**		**IMDB**		**Avg. R**
Task	NC	LP	NC	LP	NC	LP	NC	LP	(△)
Metapath2vec [18]	73.69	–	92.80	44.58	78.33	**88.86**	48.81	67.69	4.5
HetGNN [32]	82.82	**89.99**	90.43	55.73	70.61	72.37	54.78	59.24	3.9
HGConv [35]	**88.89**	82.15	93.40	57.00	92.17	63.43	63.43	64.00	3.1
HGT [11]	88.85	79.68	**93.42**	53.32	94.40	65.77	**63.63**	55.52	3.3
RGCN [33]	82.67	63.28	87.70	58.10	94.50	65.03	61.30	74.32	3.6
HeteDP (pure)	87.50	85.44	87.33	**79.72**	**97.82**	72.52	53.07	**82.07**	**2.6**
HeteDP (ϵ=0.01)	67.33	71.92	30.13	62.83	95.28	60.20	40.12	74.37	(↓ 17.9)
HeteDP (ϵ=0.1)	76.15	72.15	32.24	68.84	97.41	64.65	40.29	75.06	(↓ 14.8)
HeteDP (ϵ=1)	80.33	77.39	39.81	73.94	97.67	72.24	48.50	75.57	(↓ 10.0)
HeteSDG (ϵ=1)	89.31	80.13	35.40	80.87	98.60	64.29	54.01	74.56	(↓ 8.5)

the node representation fusion since our node is without self-loop edges; and (2) convolution-based GNNs, we choose HGConv [35], HGT [11] and RGCN [33].

Settings. We separately set the parameters of FeatADP and TopoGDP. The common parameters are set the following: training epoch to 100, epsilon ϵ from 0.01 to 1, and the probabilistic of breaking privacy protection δ to $1e-5$. For the special values, FeatADP and TopoGDP are set learning rates lr of 0.005 and 0.001, and hidden layer dimensions of 64 and 32. For the other parameters, in feature representation learning, we set the dropout of training to 0.8, the regularization coefficients to 0.001, the number of heads of the multi-headed attention mechanism K to 8, and a hyperparameter λ to 0.01. Specially, we define the meta-paths m for each node type from all possible walks, and the number of layers depends on the meta-paths m and the types of edges R in the graph. In topology learning, we set the batch size $|B|$ to 2048 and the number of negative sampling k to 5. For the baseline models, the parameters are set as the default values in their papers. To sum up, the categories of node classification and edge prediction set for each data selection follow Table 2 and the dataset split setting following VGAE [30].

5.2. Performance Comparison

We set up two downstream tasks to test the performance of our proposed method, node classification (NC) and link prediction (LP). Table 3 summarizes the performance of HeteDP and HeteSDG in the different downstream tasks and on



(a) LP on ACM. (b) LP on Amazon. (c) LP on IMDB.

Fig. 3. Ablation study of ROC-AUC scores of LP on validation set with $\epsilon = 0.01$.

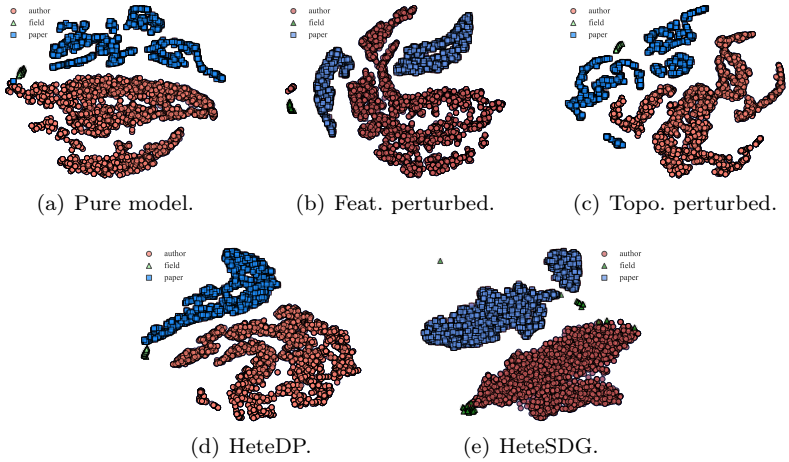
four datasets, comparing with the baseline methods, which reflects the inherent generalizability and the privacy-preserving effect.

For the node classification task, we consider the practice of unsupervised node classification [49], using negative sampling of edges for training and 2-order neighbor sampling at each iteration of validation. We use the F1 score as a classification effectiveness measure. *For the link prediction task*, we extend the sampler [49] to negative sampling on heterogeneous graph, sampling k negative pairs for each edge. Each training randomly selects a specific size of data to form batch training. The encoder of TopoGDP consists of heterogeneous convolutional layers following Eq. (16) and Eq. (17), and the link predictor calculates the scores of positive and negative sample pairs by inner product, respectively. We utilize the ROC-AUC score as an indicator to judge the performance of HeteSDG. The experimental results show that HeteDP and HeteSDG reduce the average score by 10.0% and 8.5% when privacy budget $\epsilon = 1$. This phenomenon indicates the utility of privacy preservation, and with the work of semantic-aware mechanism, it is feasible to design privacy noise more efficiently and with a lower loss of accuracy. Our designed framework has the highest average accuracy ranking in the pure state, indicating that our privacy-preserving mechanism can be implemented with a state-of-the-art learning model, ensuring the fundamental performance of the model under perturbations.

Overall, in the LP task of DBLP and IMDB, compared to the runner-up model, our proposed HeteDP (pure) improves performance over 21.62% and 7.75%. And we likewise observe that the noise of different sensitivities to each dataset brings diverse levels of influence. For example, the model accuracy improves in different magnitudes with an increasing privacy budget, such as the ROC-AUC score of Amazon is only reduced by 0.28% with $\epsilon = 1$. It shows that the generalization ability of our model is guaranteed to a certain extent, and the model can maintain the utility of the data under the influence of noise. Similar to what has been elaborated above, the ACM dataset has an accuracy reduction of about 14% on the LP task when setting the privacy-preserving strength of $\epsilon = 0.01$. It shows that our proposed privacy-preserving method can affect MIS attack enhanced by semantic relations (graph topology).

5.3. Further Analysis

In this section, we conduct an ablation study, bi-level optimization, sensitivity analysis and MIS attack verification. Some results demonstration follows HeteDP because they are a similar validation form of HeteSDG and we will not perform a new presentation of the results.



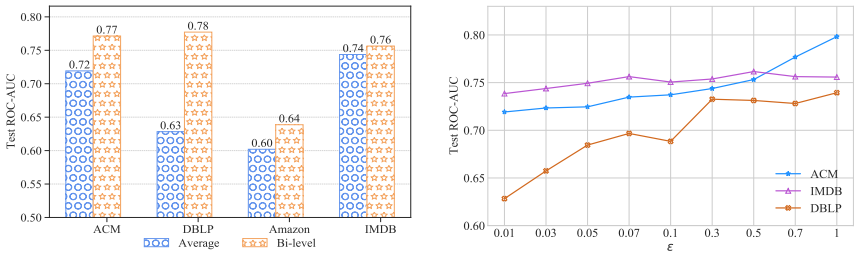
(a) Pure model. (b) Feat. perturbed. (c) Topo. perturbed.

(d) HeteDP. (e) HeteSDG.

Fig. 4. The visualization of node types on ACM.

Ablation study. We further conduct ablation experiments to assess the necessity of FeatADP and TopoGDP privacy-preserving mechanisms. We design a total of four experiments in the LP task for comparison: the first is "w/o TopoGDP" (Feature perturbed), which only protects the features of various node types by Eq. (15) in FeatADP, and the output will be used for TopoGDP; second, our expression is "w/o FeatADP" (Topology perturbed) which attains the features by aggregating the information of node neighbors and semantics, and protects the semantic relationships in the topology representation learning process with Eq. (23); the third is the version of the node feature data and the topology data are double-protected in HeteDP; Finally, the semantic-aware mechanism is added to the model named HeteSDG. Their privacy budget is 0.01 and the results are shown in Fig. 3. From the elaborated results, we can observe that both perturbations are significant, and compared with the node feature disturbance, topology perturbed has a greater impact on the model. And our training eventually reaches convergence and maintains some utility. Nevertheless, The accuracy of HeteSDG is improved faster than HeteDP. At the convergence, HeteSDG accuracy is higher, which indicates that HeteSDG has better availability and stronger adaptive capability after the usage of the semantic-aware mechanism.

Furthermore, we visualize node types to observe the utility of privacy protection in Fig. 4. It shows the 2-dimension node embedding visualization of all nodes in ACM using t-SNE [51], where the colors indicate node types. We design pure model, feature perturbed, topology perturbed, HeteDP and HeteSDG experiments, where the privacy budget is 3. The visualization from left to right generally shows increasingly tight clustering among similar nodes. For feature perturbed, we observe that node perturbation makes the spacing within the "paper" node type closer, which affects the classification effect within that node. And topology perturbed has clearer boundaries among different node types and causes a large change in the position of individual nodes even at higher ϵ. This perturbation phase has a lower impact within the node type, while the different node types become more dispersed. Compared with the pure model, this phenomenon indicates our method can distinguish different types of nodes and



(a) Bi-level optimization experiments on privacy budget allocation.

(b) Sensitivity experiments.

Fig. 5. Bi-level optimization experiments and sensitivity experiments of ROC-AUC scores on LP task.

perturb similar nodes to achieve privacy and ensure utility. Finally, compared with HeteDP, HeteSDG provides semantic information on FeatADP to make the same type of nodes more difficult to distinguish, and adds semantic regularization terms on TopoGDP to make the different types of node boundaries clearer. Since ablation studies prove that topology perturbations have a greater impact on the model, the semantic regularization term has a greater effect and the overall model accuracy is improved.

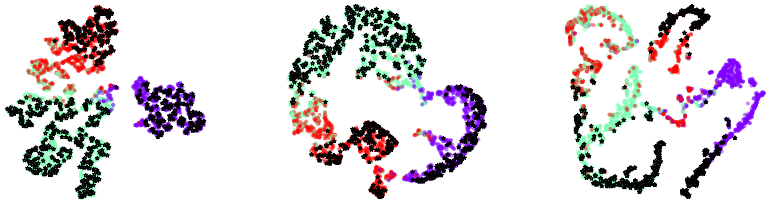
Bi-level optimization. Privacy budget allocation is an essential task in privacy protection. The aim is to reduce the probability of data being accessed by attackers, weigh the training accuracy, and try to address the problem of model performance degradation due to privacy noise. To further improve the utility of the model in privacy-preserving, we design a bi-level optimization trick to allocate the privacy budget of Gaussian noise on FeatADP and TopoGDP. We fix the topology noise and seek the optimal privacy budget allocation on node features by experiments in a specific interval according to Eq. (5) and Eq. (24). Next, we fix the amount of noise on features to find the optimal privacy budget on topology following Eq. (5) and Eq. (25). The results are shown in Fig. 5 (a). The figure compares the effect of an equally divided privacy budget and a bi-level optimized privacy budget and shows that bi-level optimization can bring better performance for the model, which achieves the purpose of the trade-off between protection power and utility.

Sensitivity Analysis. We analyze the sensitivity of the overall noise of HeteDP. Specifically, we test the influence of parameter ϵ on the LP task. We set 9 values of ϵ on ACM, IMDB and DBLP, and show in Fig. 5 (b). We observe that the ACM dataset achieves a score close to 80% at $\epsilon = 1$, which is nearly 8% higher than $\epsilon = 0.01$. IMDB, however, is not as sensitive to ϵ because its network structure is fragile, and it is harder to improve the learning ability once it is disturbed. The experiments show that different datasets have myriads of changes in sensitivities to the privacy budget due to inconsistencies in their natural data distributions, and it is necessary to find a suitable noise range to protect the model and maximize its effectiveness.

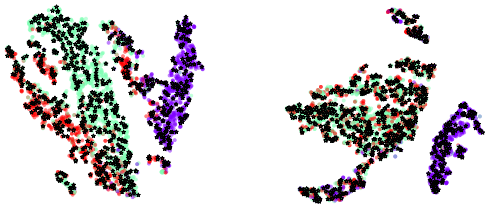
MIS Attack Verification. In the real world, the membership inference attack with semantic enhancement (MIS) models used by attackers is diverse and unknown. We use the shadow model which is similar to the structure of the original model to get the training set of the attack model and obtain the

Table 4. Micro-F1 score and Weighted-F1 score of attack accuracy.

	Dataset	original		Naive Bayesian		KNN		Decision Tree	
		M-F1	W-F1	M-F1	W-F1	M-F1	W-F1	M-F1	W-F1
ACM	HGConv	90.82	90.34	74.15	71.31	77.13	71.09	71.86	69.28
	HGT	87.91	85.21	79.22	71.52	77.73	71.29	72.86	69.44
	RGCN	83.48	83.02	75.34	71.10	75.74	70.02	68.78	67.37
	HeteSDG ($\epsilon = 0.01$)	**81.14**	**80.59**	**44.13**	**40.99**	**45.22**	**45.84**	**45.74**	**43.19**
	HeteSDG ($\epsilon = 1$)	88.31	88.37	46.62	47.13	45.52	45.75	47.61	48.12
IMDB	HGConv	57.62	57.10	41.05	36.83	50.99	52.89	50.09	52.10
	HGT	55.56	54.95	51.88	53.35	48.90	50.99	49.40	51.42
	RGCN	51.50	51.70	41.94	41.04	49.10	51.10	50.59	52.64
	HeteSDG ($\epsilon = 0.01$)	**42.92**	**42.73**	**30.41**	**16.34**	**32.89**	**25.76**	**40.85**	**40.06**
	HeteSDG ($\epsilon = 1$)	52.80	52.70	34.89	26.76	33.59	26.38	42.84	42.72



(a) HGConv (b) HGT (c) RGCN



(d) HeteSDG ($\epsilon = 1$) (e) HeteSDG ($\epsilon = 0.01$)

Fig. 6. "paper" embedding visualization of ACM.

results of the target model query to get the test set of the attack model [52]. To make the attack experiments more representative, we utilize three widely familiar classical inference models as attack models: Naive Bayesian, KNN, and Decision Trees. We selected the convolution-based aggregation methods in the baseline (HGConv, HGT, RGCN) as the comparison models, i.e., the meta-paths-based design methods are not considered in the design of the attack models, since this is a semantic enhancement-like and an independent process from the model training. Table 4 shows the attack accuracy of each attacker who conducts "paper" node type inference on ACM and "movie" node type inference on IMDB. The experimental results expose that the attack models obtain more information on the baseline model but achieve poorer inference results on our model. In particular, HeteSDG achieves the lowest attack accuracy at stronger privacy-preserving strengths. It shows that our method is able to resist inference attacks while maintaining the representation learning capability. In addition, we show "paper" embedding visualization of ACM in Fig. 6 and we also obtain the same conclusion as above. An additional gain is that the perturbation ability of privacy

budget strength on representation learning is demonstrated. In general, HeteSGD can guarantee privacy while ensuring accuracy for downstream tasks. For example, with $\epsilon = 1$, the classification accuracy for the paper is 88.31%, which is higher than HGT and RGCN. And the classification visualization is well bounded which indicates the availability, while the attack accuracy decreases at least by 40.7% which shows the power of privacy protection.

6. Conclusion

In this work, we propose HeteSDG, a novel heterogeneous graph neural network with semantic-aware differential privacy guarantees, i.e. we propose a two-stage privacy-preserving mechanism based on differential privacy, capable of adapting to the high-order features and topological complexity caused by semantics. For multi-nodes and multi-relationships, we learn the representation distribution and aggregation of nodes on each relationship through multi-relational convolutional layers and adapt to various downstream tasks through unsupervised learning. Considering that nodes and topology are vulnerable to MIS attack in heterogeneous graph scenarios, we perturb the node features and the topology structure, respectively. In particular, we design a unique semantic-aware debias mechanism to guide more accurate noise generation and enhance the utility of the privacy-preserving model. Then, we balance the privacy budget allocation of the node feature and the topology structures, and achieve higher performance by bi-level optimization. Comprehensive experiments on four datasets demonstrate the privacy-preserving capability and adaptability of HeteSDG, and the MIS attack experiments on three basic attack models show that our model produces resistance against MIS attacks with guaranteed accuracy utility. We hope that our work could bring some inspiration to privacy protection in more complex graph data.

Acknowledgment

This paper was supported by the National Natural Science Foundation of China (No. 62162005), Guangxi Science and Technology Major Project (No. AA22068070), National Natural Science Foundation of China (No. U21A20474), Research Fund of Guangxi Key Lab of Multi-source Information Mining & Security (No. 19-A-02-01), Guangxi 1000-Plan of Training Middle-aged/Young Teachers in Higher Education Institutions, Guangxi "Bagui Scholar" Teams for Innovation and Research Project, and Guangxi Collaborative Innovation Center of Multisource Information Integration and Intelligent Processing.

References

[1] Yu He, Yangqiu Song, Jianxin Li, Cheng Ji, Jian Peng, and Hao Peng. Hetespaceywalk: A heterogeneous spacey random walk for heterogeneous information network embedding. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*, pages 639–648. ACM, 2019.

[2] Yuxiao Dong, Ziniu Hu, Kuansan Wang, Yizhou Sun, and Jie Tang. Heterogeneous network representation learning. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 4861–4867. ijcai.org, 2020.

[3] Jibing Gong, Shen Wang, Jinlong Wang, Wenzheng Feng, Hao Peng, Jie Tang, and Philip S. Yu. Attentional graph convolutional networks for knowledge concept recommendation in moocs in a heterogeneous view. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 79–88. ACM, 2020.

[4] Chong Chen, Weizhi Ma, Min Zhang, Zhaowei Wang, Xiuqiang He, Chenyang Wang, Yiqun Liu, and Shaoping Ma. Graph heterogeneous multi-relational recommendation. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 3958–3966. AAAI Press, 2021.

[5] Qingyun Sun, Jianxin Li, Hao Peng, Jia Wu, Yuanxing Ning, Philip S. Yu, and Lifang He. SUGAR: subgraph neural network with reinforcement pooling and self-supervised mutual information mechanism. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*, pages 2081–2091. ACM / IW3C2, 2021.

[6] Jianxin Li, Xingcheng Fu, Hao Peng, Senzhang Wang, Shijie Zhu, Qingyun Sun, Philip S. Yu, and Lifang He. A robust and generalized framework for adversarial graph embedding. *arXiv preprint arXiv:2105.10651*, 2021.

[7] Qingyun Sun, Jianxin Li, Hao Peng, Jia Wu, Xingcheng Fu, Cheng Ji, and Philip S. Yu. Graph structure learning with variational information bottleneck. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 4165–4174. AAAI Press, 2022.

[8] Chen Li, Hao Peng, Jianxin Li, Lichao Sun, Lingjuan Lyu, Lihong Wang, Philip S. Yu, and Lifang He. Joint stance and rumor detection in hierarchical heterogeneous graph. *IEEE Trans. Neural Networks Learn. Syst.*, 33(6):2530–2542, 2022.

[9] Qingyun Sun, Jianxin Li, Haonan Yuan, Xingcheng Fu, Hao Peng, Cheng Ji, Qian Li, and Philip S. Yu. Position-aware structure learning for graph topology-imbalance by relieving under-reaching and over-squashing. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, pages 1848–1857. ACM, 2022.

[10] Xinyu Fu, Jiani Zhang, Ziqiao Meng, and Irwin King. MAGNN: metapath aggregated graph neural network for heterogeneous graph embedding. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 2331–2341. ACM / IW3C2, 2020.

[11] Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun. Heterogeneous graph transformer. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 2704–2710. ACM / IW3C2, 2020.

[12] Jianxin Li, Hao Peng, Yuwei Cao, Yingtong Dou, Hekai Zhang, Philip S. Yu, and Lifang He. Higher-order attribute-enhancing heterogeneous graph neural networks. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[13] Linhao Luo, Yixiang Fang, Xin Cao, Xiaofeng Zhang, and Wenjie Zhang. Detecting communities from heterogeneous graphs: A context path-based graph neural network model. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*, pages 1170–1180. ACM, 2021.

[14] Jianan Zhao, Xiao Wang, Chuan Shi, Binbin Hu, Guojie Song, and Yanfang Ye. Heterogeneous graph structure learning for graph neural networks. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in*

Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021, pages 4697–4705. AAAI Press, 2021.

[15]Junliang Yu, Hongzhi Yin, Jundong Li, Qinyong Wang, Nguyen Quoc Viet Hung, and Xiangliang Zhang. Self-supervised multi-channel hypergraph convolutional network for social recommendation. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*, pages 413–424. ACM / IW3C2, 2021.

[16]Yang Liu, Chen Liang, Xiangnan He, Jiaying Peng, Zibin Zheng, and Jie Tang. Modelling high-order social relations for item recommendation. *IEEE Trans. Knowl. Data Eng.*, 34(9):4385–4397, 2022.

[17]Tao-Yang Fu, Wang-Chien Lee, and Zhen Lei. Hin2vec: Explore meta-paths in heterogeneous information networks for representation learning. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017*, pages 1797–1806. ACM, 2017.

[18]Yuxiao Dong, Nitesh V. Chawla, and Ananthram Swami. metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*, pages 135–144. ACM, 2017.

[19]Huiting Hong, Hantao Guo, Yucheng Lin, Xiaoqing Yang, Zang Li, and Jieping Ye. An attention-based graph neural network for heterogeneous structural learning. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 4132–4139. AAAI Press, 2020.

[20]Huaxin Li, Qingrong Chen, Haojin Zhu, Di Ma, Hong Wen, and Xuemin Sherman Shen. Privacy leakage via de-anonymization and aggregation in heterogeneous social networks. *IEEE Trans. Dependable Secur. Comput.*, 17(2):350–362, 2020.

[21]Jiachun Li and Guoqian Chen. A personalized trajectory privacy protection method. *Comput. Secur.*, 108:102323, 2021.

[22]Behnaz Bostanipour and George Theodorakopoulos. Joint obfuscation of location and its semantic information for privacy protection. *Comput. Secur.*, 107:102310, 2021.

[23]Yanhui Li, Xin Cao, Ye Yuan, and Guoren Wang. Privsem: Protecting location privacy using semantic and differential privacy. *World Wide Web*, 22(6):2407–2436, 2019.

[24]Behnaz Bostanipour and George Theodorakopoulos. Joint obfuscation of location and its semantic information for privacy protection. *Comput. Secur.*, 107:102310, 2021.

[25]Mariana Cunha, Ricardo Mendes, and João P. Vilela. A survey of privacy-preserving mechanisms for heterogeneous data types. *Comput. Sci. Rev.*, 41:100403, 2021.

[26]Huaxin Li, Qingrong Chen, Haojin Zhu, Di Ma, Hong Wen, and Xuemin Sherman Shen. Privacy leakage via de-anonymization and aggregation in heterogeneous social networks. *IEEE Trans. Dependable Secur. Comput.*, 17(2):350–362, 2020.

[27]Shijie Zhang, Hongzhi Yin, Tong Chen, Zi Huang, Lizhen Cui, and Xiangliang Zhang. Graph embedding for recommendation against attribute inference attacks. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*, pages 3002–3014. ACM / IW3C2, 2021.

[28]Carl Yang, Haonan Wang, Ke Zhang, Liang Chen, and Lichao Sun. Secure deep graph generation with link differential privacy. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pages 3271–3278. ijcai.org, 2021.

[29]Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4):211–407, 2014.

[30]Thomas N. Kipf and Max Welling. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*, 2016.

[31]Yuecen Wei, Xingcheng Fu, Qingyun Sun, Hao Peng, Jia Wu, Jinyan Wang, and Xianxian Li. Heterogeneous graph neural network for privacy-preserving recommendation. *arXiv preprint arXiv:2210.00538*, 2022.

[32]Chuxu Zhang, Dongjin Song, Chao Huang, Ananthram Swami, and Nitesh V. Chawla. Heterogeneous graph neural network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 793–803. ACM, 2019.

[33]Michael Sejr Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *The Semantic Web - 15th International Conference, ESWC 2018, Heraklion, Crete, Greece,*

June 3-7, 2018, Proceedings, volume 10843 of *Lecture Notes in Computer Science*, pages 593–607. Springer, 2018.

[34]Rianne van den Berg, Thomas N. Kipf, and Max Welling. Graph convolutional matrix completion. *arXiv preprint arXiv:1706.02263*, 2017.

[35]Le Yu, Leilei Sun, Bowen Du, Chuanren Liu, Weifeng Lv, and Hui Xiong. Hybrid micro/macro level convolution for heterogeneous graph learning. *arXiv preprint arXiv:2012.14722*, 2020.

[36]Huiting Hong, Hantao Guo, Yucheng Lin, Xiaoqing Yang, Zang Li, and Jieping Ye. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 4132–4139. AAAI Press, 2020.

[37]Le Wu, Junwei Li, Peijie Sun, Richang Hong, Yong Ge, and Meng Wang. Diffnet++: A neural influence and interest diffusion network for social recommendation. *IEEE Trans. Knowl. Data Eng.*, 34(10):4753–4766, 2022.

[38]Fengli Xu, Jianxun Lian, Zhenyu Han, Yong Li, Yujian Xu, and Xing Xie. Relation-aware graph convolutional networks for agent-initiated social e-commerce recommendation. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*, pages 529–538. ACM, 2019.

[39]Mingxuan Yuan, Lei Chen, and Philip S. Yu. Personalized privacy protection in social networks. *Proc. VLDB Endow.*, 4(2):141–150, 2010.

[40]Elena Zheleva and Lise Getoor. Preserving the privacy of sensitive relationships in graph data. In *Privacy, Security, and Trust in KDD, First ACM SIGKDD International Workshop, PinKDD 2007, San Jose, CA, USA, August 12, 2007, Revised Selected Papers*, volume 4890 of *Lecture Notes in Computer Science*, pages 153–171. Springer, 2007.

[41]Kun Liu and Evimaria Terzi. Towards identity anonymization on graphs. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008*, pages 93–106. ACM, 2008.

[42]Martín Abadi, Andy Chu, Ian J. Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, pages 308–318. ACM, 2016.

[43]Sina Sajadmanesh and Daniel Gatica-Perez. Locally private graph neural networks. In Yongdae Kim, Jong Kim, Giovanni Vigna, and Elaine Shi, editors, *CCS '21: 2021 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, Republic of Korea, November 15 - 19, 2021*, pages 2130–2145. ACM, 2021.

[44]Iyiola E. Olatunji, Thorben Funke, and Megha Khosla. Releasing graph neural networks with differential privacy guarantees. *arXiv preprint arXiv:2109.08907*, 2021.

[45]Sahel Torkamani, Javad B. Ebrahimi, Parastoo Sadeghi, Rafael G. L. D'Oliveira, and Muriel Médard. Heterogeneous differential privacy via graphs. In *IEEE International Symposium on Information Theory, ISIT 2022, Espoo, Finland, June 26 - July 1, 2022*, pages 1623–1628. IEEE, 2022.

[46]Cynthia Dwork. Differential privacy. In *Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II*, volume 4052 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2006.

[47]Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam D. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography, Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006, Proceedings*, volume 3876 of *Lecture Notes in Computer Science*, pages 265–284. Springer, 2006.

[48]Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017.

[49]William L. Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 1024–1034, 2017.

[50]Luca Franceschi, Michele Donini, Paolo Frasconi, and Massimiliano Pontil. Forward and reverse gradient-based hyperparameter optimization. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1165–1173. PMLR, 2017.

[51]Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.
[52]Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pages 3–18. IEEE Computer Society, 2017.

Author Biographies

Yuecen Wei is currently an M.S. candidate at the School of Computer Science and Engineering, Guangxi Normal University, Guilin, China. Her research interests include graph representation learning, privacy protection and social network analysis. She has published one paper on ICDM.

Xingcheng Fu is currently a Ph.D. candidate at the Beijing Advanced Innovation Center for Big Data and Brain Computing in Beihang University. His research interests include graph representation learning, complex network and social network analysis. He has published several papers on Web Conference (WWW), AAAI, ICDM, CIKM, WSDM, etc.

Dongqi Yan is currently an M.S. candidate at the School of Computer Science and Engineering, Guangxi Normal University, Guilin, China. His research interests include graph representation learning, privacy protection and federated learning.

Qingyun Sun is currently an Assistant Professor at the Beijing Advanced Innovation Center for Big Data and Brain Computing in Beihang University. Her research interests include data mining and graph representation learning. Her research interests include machine learning and graph mining. She has published several papers on Web Conference (WWW), AAAI, ICDM, CIKM, WSDM, etc.

Hao Peng is currently an Associate Professor at the Beijing Advanced Innovation Center for Big Data and Brain Computing in Beihang University. His research interests include representation learning, data mining, and reinforcement learning. Dr. Peng has published over 80+ research papers in top-tier journals and conferences, including the IEEE TPAMI, TKDE, TPDS, TNNLS, TASLP, JAIR, ACM TOIS, TKDD, and Web Conference. He is the Associate Editor of the International Journal of Machine Learning and Cybernetics.

Jia Wu is an ARC DECRA Fellow at the School of Computing, Macquarie University, Sydney, Australia. He received the Ph.D. from the University of Technology Sydney, Australia. His current research interests include data mining and machine learning. He has published 100+ refereed research papers in the above areas such as TPAMI, TKDE, TNNLS, TMM, NIPS, KDD, ICDM, IJCAI, AAAI and WWW. Dr. Wu was the recipient of SDM'18 Best Paper Award in Data Science Track and IJCNN'17 Best Student Paper Award. He currently serves as an Associate Editor of ACM TKDD. He is a Senior Member of IEEE.

Jinyan Wang is currently a professor at the School of Computer Science and Engineering, Guangxi Normal University, Guilin, China. She received the B.Sc., M.Sc. and Ph.D. degrees from the School of Computer Science and Information Technology, Northeast Normal University, Changchun, China, in 2005, 2008 and 2011, respectively. Her research interests include machine learning and information security.

Xianxian Li is currently a professor at the School of Computer Science and Engineering, Guangxi Normal University, Guilin, China. He received the Ph.D. degree from the School of Computer Science and Engineering, Beihang University, Beijing, China, in 2002. He worked as a professor at Beihang University during 2003-2010. His research interests include information security and machine learning.

Correspondence and offprint requests to: Jinyan Wang, Guangxi Key Lab of Multi-source Information Mining & Security and School of Computer Science and Engineering, Guangxi Normal University, Guilin, 541004, Guangxi, China. Email: wangjy612@gxnu.edu.cn