Secure Deep Graph Generation with Link Differential Privacy: Appendices

Carl Yang[1*], **Haonan Wang**[2†], **Ke Zhang**[3†], **Liang Chen**[4], **Lichao Sun**[5]
[1]Emory University
[2]University of Illinois at Urbana Champaign
[3]University of Hong Kong
[4]Sun Yat-sen University
[5]Lehigh University
j.carlyang@emory.edu, haonan3@illinois.edu, cszhangk@connect.hku.hk
chenliang6@mail.sysu.edu.cn, lis221@lehigh.edu

A APPENDIX: Proofs for Theorem 1

In this appendix, we provide proofs for Theorem 1, and derive Corollary 1.1 and Corollary 1.2. Theorem 1 indicates the link privacy protection achieved through updating model's parameters with clipped and noised gradient (latter referred to as DP learning) for link reconstruction based graph generation models. Corollary 1.1 and Corollary 1.2 derived from Theorem 1 support us to guarantee (ε, δ)-edge-DP for DPGVAE and DPGGAN with DP learning in Theorem 1.

The proof for Theorem 1 is divided into three steps. We first briefly introduce the definition of the moment accountant privacy analysis and respective properties in Section A.1, for it being the fundamentals of our proof. Note that in [Abadi *et al.*, 2016], DPSGD is originally designed for classical machine learning tasks, such as image classification. Therefore, in Section A.2, we leverage moment accountant to conduct the extended privacy analysis of DPSGD for general types of data and loss functions. Then in Section A.3, we apply the conclusion from Section A.2 on graph data and the link reconstruction loss function to derive the theoretical analysis over edge-DP achieved by link reconstruction based graph generation models and finish our proof for Theorem 1. Following the conclusion in Theorem 1, we tune gradient representations to certain gradient functions leveraged in training DPGVAE decoder and DPGGAN generator to derive Corollary 1.1 and Corollary 1.2, as the theoretical support for the (ε, δ)-edge-DP held by respective models.

A.1 Moment Accountant

Our proof for Theorem 1 is mainly based on moment accountant [Abadi *et al.*, 2016]. The definition of moment accountant and the properties leveraged in our proof are listed below.

Definition 1. *Let $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ be a randomized mechanism and d, d' a pair of adjacent databases. Let aux denote an auxiliary input. For an outcome $o \in \mathcal{R}$, the privacy loss at o is defined as:*

$$c\left(o; \mathcal{M}, aux, d, d'\right) \triangleq \log \frac{\Pr[\mathcal{M}(aux, d) = o]}{\Pr\left[\mathcal{M}\left(aux, d'\right) = o\right]} \quad (1)$$

The privacy loss random variable $C\left(\mathcal{M}, aux, d, d'\right)$ is defined as $c\left(\mathcal{M}(d); \mathcal{M}, aux, d, d'\right)$, i.e.the random variable defined by evaluating the privacy loss at an outcome sampled from $\mathcal{M}(d)$.

Definition 2. *Let $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ be a randomized mechanism and d, d' a pair of adjacent databases. Let aux denote an auxiliary input. The moments accountant is defined as:*

$$\alpha_{\mathcal{M}}(\lambda) \triangleq \max_{aux, d, d'} \alpha_{\mathcal{M}}\left(\lambda; aux, d, d'\right) \quad (2)$$

where $\alpha_{\mathcal{M}}\left(\lambda; aux, d, d'\right) \triangleq \log \mathbb{E}\left[\exp\left(\lambda C\left(\mathcal{M}, aux, d, d'\right)\right)\right]$ is the moment generating function of the privacy loss random variable.

The following properties of the moments accountant are proved in [Abadi *et al.*, 2016].

Property 2.1. *[Composability] Suppose that a mechanism $\mathcal{M}$ consists of a sequence of adaptive mechanisms $\mathcal{M}_1, \ldots, \mathcal{M}_k$ where $\mathcal{M}_i : \prod_{j=1}^{i-1} \mathcal{R}_j \times \mathcal{D} \to \mathcal{R}_i$. Then, for any output sequence $o_1, \ldots, o_{k-1}$ and any λ, we have*

$$\alpha_{\mathcal{M}}\left(\lambda; d, d'\right) = \sum_{i=1}^{k} \alpha_{\mathcal{M}_i}\left(\lambda; o_1, \ldots, o_{i-1}, d, d'\right) \quad (3)$$

where $\alpha_{\mathcal{M}}$ is conditioned on $\mathcal{M}_i$'s output being o_i for $i < k$.

Property 2.2. *[Tail bound] For any $\varepsilon > 0$, the mechanism $\mathcal{M}$ is (ε, δ)-DP for*

$$\delta = \min_{\lambda} \exp\left(\alpha_{\mathcal{M}}(\lambda) - \lambda\varepsilon\right) \quad (4)$$

A.2 The Generalized Privacy Analysis of DPSGD

To achieve (ε, δ)-edge-DP for graph data, we exploit DPSGD [Abadi *et al.*, 2016] with necessary adaptions according to the special nature of graph data compared to other types of data (*e.g.*, images), for which DPSGD was originally designed. The original DPSGD only provides DP proof for gradient function f clipped by C with its ℓ_2-norm sensitivity as $\Delta_2 f = 1 \cdot C = C$. For classical tasks of machine learning like image classification, $\Delta_2 f = C$ is obvious. However, in a more complex task like graph learning, depending on the chosen measurement, the influence for the output induced by a minor change in the training dataset varies. To explore the

*Corresponding Author
†Equal Contribution

potential of DPSGD with customized machine learning tasks, we further prove the privacy performance of DPSGD with a gradient function f with ℓ_2-norm sensitivity $\Delta_2 f = s$.

Therefore, to prepare for the proof for Theorem 1, we first leverage moments accountant to derive the upper bound of privacy loss for a Gaussian Mechanism as below.

Lemma 1. *Suppose that $f : D \to \mathbb{R}^p$ with $\|f(\cdot)\|_2 \leq s$. Let $\sigma \geq s$ and let J be a sample from $[n]$ where each $i \in [n]$ is chosen independently with probability $q < \frac{s}{16\sigma}$. Then for any positive integer $\lambda \leq \frac{\sigma^2}{s^2} \ln \frac{s}{q\sigma}$, the Gaussian Mechanism $\mathcal{M}(d) = \sum_{i \in J} f(d_i) + \mathcal{N}\left(0, \sigma^2 \mathbf{I}\right)$ satisfies*

$$\alpha_{\mathcal{M}}(\lambda) \leq \frac{s^2 q^2 \lambda(\lambda+1)}{(1-q)\sigma^2} + O\left(s^3 q^3 \lambda^3 / \sigma^3\right) \tag{5}$$

Proof. Fix d' and let $d = d' \cup \{d_n\}$. Without loss of generality, we assume $f(d_n) = s \cdot \mathbf{e}_1$ and $\sum_{i \in J \setminus [n]} f(d_i) = \mathbf{0}$. Thus $\mathcal{M}(d)$ and $\mathcal{M}(d')$ are distributed identically except for the first coordinate and hence we have a one-dimensional problem. Let μ_0 denote the pdf of $\mathcal{N}\left(0, \sigma^2\right)$ and let μ_s denote the pdf of $\mathcal{N}\left(s, \sigma^2\right)$. We have

$$\begin{aligned} &\mathcal{M}(d') \sim \mu_0, \\ &\mathcal{M}(d) \sim \mu \triangleq (1-q)\mu_0 + q\mu_s. \end{aligned} \tag{6}$$

We want to show that

$$\begin{aligned} &\mathbb{E}_{z \sim \mu}\left[\left(\mu(z)/\mu_0(z)\right)^{\lambda}\right] \leq \alpha, \\ \text{and } &\mathbb{E}_{z \sim \mu_0}\left[\left(\mu_0(z)/\mu(z)\right)^{\lambda}\right] \leq \alpha, \end{aligned} \tag{7}$$

where α is a value to be determined. We will use the same method as in [Abadi *et al.*, 2016] to prove both bounds. Assume we have two distributions ν_0 and ν_s, and we wish to bound

$$\mathbb{E}_{z \sim \nu_0}\left[\left(\nu_0(z)/\nu_s(z)\right)^{\lambda}\right] = \mathbb{E}_{z \sim \nu_s}\left[\left(\nu_0(z)/\nu_s(z)\right)^{\lambda+1}\right]. \tag{8}$$

Leveraging binomial expansion, we obtain

$$\begin{aligned} &\mathbb{E}_{z \sim \nu_s}\left[\left(\nu_0(z)/\nu_s(z)\right)^{\lambda+1}\right] \\ =&\mathbb{E}_{z \sim \nu_s}\left[\left(1 + \left(\nu_0(z) - \nu_s(z)\right)/\nu_s(z)\right)^{\lambda+1}\right] \\ =&\mathbb{E}_{z \sim \nu_s}\left[\left(1 + \left(\nu_0(z) - \nu_s(z)\right)/\nu_s(z)\right)^{\lambda+1}\right] \\ =&\sum_{t=0}^{\lambda+1} \binom{\lambda+1}{t} \mathbb{E}_{z \sim \nu_s}\left[\left(\left(\nu_0(z) - \nu_s(z)\right)/\nu_s(z)\right)^{t}\right]. \end{aligned} \tag{9}$$

The first term in Eq. (9) is 1, and the second term is

$$\begin{aligned} &(\lambda+1)\mathbb{E}_{z \sim \nu_s}\left[\frac{\nu_0(z) - \nu_s(z)}{\nu_s(z)}\right] \\ &= \int_{-\infty}^{+\infty} \nu_s(z) \frac{\nu_0(z) - \nu_s(z)}{\nu_s(z)} \mathrm{d}z \\ &= (\lambda+1)\int_{-\infty}^{+\infty} \nu_0(z)\mathrm{d}z - \int_{-\infty}^{+\infty} \nu_s(z)\mathrm{d}z \\ &= (\lambda+1)(1-1) = 0. \end{aligned} \tag{10}$$

Regarding conditions stated in the lemma, for both cases, where $\nu_0 = \mu, \nu_1 = \mu_0$ and $\nu_0 = \mu_0, \nu_1 = \mu$, the third term is bounded by $q^2\lambda(\lambda+1)/(1-q)\sigma^2$ and this bound dominates the sum of the remaining terms. We provide the proof for the case of $(\nu_0 = \mu_0, \nu_s = \mu)$, and the proof of the other case is similar.

To upper bound the third term in 9, we note that $\mu(z) \geq (1-q)\mu_0(z)$, and write

$$\begin{aligned} &\mathbb{E}_{z \sim \mu}\left[\left(\frac{\mu_0(z) - \mu(z)}{\mu(z)}\right)^2\right] \\ &= q^2 \mathbb{E}_{z \sim \mu}\left[\left(\frac{\mu_0(z) - \mu_s(z)}{\mu(z)}\right)^2\right] \\ &= q^2 \int_{-\infty}^{+\infty} \frac{\left(\mu_0(z) - \mu_s(z)\right)^2}{\mu(z)} \mathrm{d}z \\ &\leq \frac{q^2}{1-q} \int_{-\infty}^{+\infty} \frac{\left(\mu_0(z) - \mu_s(z)\right)^2}{\mu_0(z)} \mathrm{d}z \\ &= \frac{q^2}{1-q} \mathbb{E}_{z \sim \mu_0}\left[\left(\frac{\mu_0(z) - \mu_s(z)}{\mu_0(z)}\right)^2\right]. \end{aligned} \tag{11}$$

Recalling the definition of μ_0 and the normal distribution, we have

$$\begin{aligned} &\mathbb{E}_{z \sim \mu_0}\left[\left(\frac{\mu_0(z) - \mu_1(z)}{\mu_0(z)}\right)^2\right] \\ &= \mathbb{E}_{z \sim \mu_0}\left[\left(1 - \exp\left(\frac{2sz - s^2}{2\sigma^2}\right)\right)^2\right] \\ &= 1 - 2\mathbb{E}_{z \sim \mu_0}\left[\exp\left(\frac{2sz - s^2}{2\sigma^2}\right)\right] + \mathbb{E}_{z \sim \mu_0}\left[\exp\left(\frac{4sz - 2s^2}{2\sigma^2}\right)\right]. \end{aligned} \tag{12}$$

For the second term in Eq. (12) $\mathbb{E}_{z \sim \mu_0}\left[\exp\left(\frac{2sz - s^2}{2\sigma^2}\right)\right]$, we have

$$\begin{aligned} &\mathbb{E}_{z \sim \mu_0}\left[\exp\left(\frac{2sz - s^2}{2\sigma^2}\right)\right] \\ &= \int_{-\infty}^{+\infty} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(z-s)^2}{2\sigma^2}\right) \mathrm{d}z = 1. \end{aligned} \tag{13}$$

For the third term in Eq. (12), we have

$$\begin{aligned} &\mathbb{E}_{z \sim \mu_0}\left[\exp\left(\frac{4sz - 2s^2}{2\sigma^2}\right)\right] \\ &= \exp\left(\frac{s^2}{\sigma^2}\right) \int_{-\infty}^{+\infty} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(z-2s)^2}{2\sigma^2}\right) \mathrm{d}z \\ &= \exp\left(\frac{s^2}{\sigma^2}\right). \end{aligned} \tag{14}$$

Thus, for Eq. (12), we have

$$\mathbb{E}_{z \sim \mu_0}\left[\left(\frac{\mu_0(z) - \mu_1(z)}{\mu_0(z)}\right)^2\right] = \exp\left(s^2/\sigma^2\right) - 1. \tag{15}$$

Hence, the third term in the binomial expansion of Eq. (9) is

$$\begin{aligned}\binom{1+\lambda}{2}\mathbb{E}_{z\in\mu}\left[\left(\frac{\mu_0(z)-\mu(z)}{\mu(z)}\right)^2\right]\\ \leq \frac{\lambda(\lambda+1)q^2}{2(1-q)}\left(\exp(\frac{s^2}{\sigma^2})-1\right)\end{aligned} \tag{16}$$

For $\sigma \geq s$, it is easy to get $\exp(\frac{s^2}{\sigma^2})-1 \leq \frac{2s^2}{\sigma^2}$. Therefore, we retrieve that

$$\binom{1+\lambda}{2}\mathbb{E}_{z\in\mu}\left[\left(\frac{\mu_0(z)-\mu(z)}{\mu(z)}\right)^2\right] \leq \frac{\lambda(\lambda+1)q^2s^2}{(1-q)\sigma^2}. \tag{17}$$

By standard calculus, we get $|\mu_0(z)-\mu_s(z)| = \left|\int_{z-s}^{z}\mu_0'(z)\mathrm{d}z\right|$. Note that $\mu_0'(z)$ is monotonically decreasing in $(-\infty,+\infty)$. Thus, to bound the remaining terms, we derive

$$\begin{aligned}\forall z \leq 0 : |\mu_0(z)-\mu_s(z)| &\leq -s(z-s)\mu_s(z)/\sigma^2\\ \forall z \geq s : |\mu_0(z)-\mu_s(z)| &\leq zs\mu_0(z)/\sigma^2\\ \forall 0 \leq z \leq s : |\mu_0(z)-\mu_s(z)| &\leq \mu_0(z)\left(\exp\left(s^2/2\sigma^2\right)-1\right)\\ &\leq s^2\mu_0(z)/\sigma^2.\end{aligned} \tag{18}$$

We can then write

$$\begin{aligned}&\mathbb{E}_{z\sim\mu}\left[\left(\frac{\mu_0(z)-\mu(z)}{\mu(z)}\right)^t\right]\\ &\leq \int_{-\infty}^{0}\mu(z)\left|\left(\frac{\mu_0(z)-\mu(z)}{\mu(z)}\right)^t\right|\mathrm{d}z\\ &+\int_{0}^{s}\mu(z)\left|\left(\frac{\mu_0(z)-\mu(z)}{\mu(z)}\right)^t\right|\mathrm{d}z\\ &+\int_{s}^{+\infty}\mu(z)\left|\left(\frac{\mu_0(z)-\mu(z)}{\mu(z)}\right)^t\right|\mathrm{d}z.\end{aligned} \tag{19}$$

We consider these terms individually. We repeatedly make use of three observations: (1) $\mu_0-\mu = q\left(\mu_0-\mu_s\right)$,(2)$\mu \geq (1-q)\mu_0$, (3)$\mu \geq q\mu_s$, and (4) $\mathbb{E}_{\mu_0}\left[|z|^t\right] \leq \sigma^t(t-1)!!$. The first term can then be bounded by

$$\begin{aligned}&\frac{q^t}{(1-q)^{t-1}\sigma^{2t}}\int_{-\infty}^{0}\mu_0(z)|z-1|^t\mathrm{d}z\\ &\leq \int_{-\infty}^{0}q\mu_s\left|\left(\frac{\mu_0-\mu_s}{\mu_s}\right)^t\right|\mathrm{d}z\\ &\leq \frac{qs^t}{\sigma^{2t}}\int_{-\infty}^{0}\mu_s\left|(z-s)^t\right|\mathrm{d}z\\ &\leq \frac{qs^t(t-1)!!}{2\sigma^t}.\end{aligned} \tag{20}$$

Then the second term is at most

$$\begin{aligned}&\frac{q^t}{(1-q)^t}\int_0^s\mu(z)\left|\left(\frac{\mu_0(z)-\mu_1(z)}{\mu_0(z)}\right)^t\right|\mathrm{d}z\\ &\leq \frac{q^t}{(1-q)^t}\int_0^s\mu(z)\left|\left(s^2/\sigma^2\right)^t\right|\mathrm{d}z\\ &\leq \frac{q^ts^{2t}}{(1-q)^t\sigma^{2t}}.\end{aligned} \tag{21}$$

Similarly, the third term is at most

$$\frac{q^ts^t}{(1-q)^{t-1}\sigma^{2t}}\int_s^{+\infty}\mu_0(z)\left|z^t\right|\mathrm{d}z \leq \frac{q^ts^t(t-1)!!}{(1-q)^{t-1}\sigma^t}. \tag{22}$$

Under the assumptions on q,σ, and λ, it is easy to check that the three terms, and their sum, drop off geometrically fast in t for $t>3$. Hence the binomial expansion (5) is dominated by the $t=3$ term, which is $O\left(s^3q^3\lambda^3/\sigma^3\right)$. Therefore, the lemma is proved. □

With Lemma 1, we retrieve the upper bound of privacy loss of the Gaussian Mechanism. Hence, based on Lemma 1 and Property 2.1, we provide the generalized privacy analysis of DPSGD with different learning tasks, which iteratively performs multiple times of the Gaussian Mechanism.

Lemma 2. *Suppose that $f : D \to \mathbb{R}^p$ with $\|f(\cdot)\|_2 \leq s$. Let J be a sample from $[N]$ that each $i \in [N]$ is chosen independently in probability $q = |J|/N$, given the number of steps T, for any $c_0 \in (0,1)$, there exist explicit constants c_1 and c_2 that with any $\varepsilon < c_1q^2T$, iteratively computing T times of $\mathcal{M}(d)$ in Lemma 1 attains it with (ε,δ)-DP for any $\delta > 0$ if we choose*

$$\sigma \geq c_2\frac{qs\sqrt{T\log(1/\delta)}}{\varepsilon}, \tag{23}$$

where $c_1 \geq \frac{1}{c_0}\log\frac{s}{q\sigma}$ and $c_2 \leq \frac{1}{\sqrt{c_0(1-c_0)}}$ for any $c_0 \in (0,1)$.

Proof. Assume for now that σ,λ satisfy the conditions in Lemma 1. After T times of iteration, with Property 2.1 we derive that $\alpha(\lambda) \leq Tq^2s^2\lambda^2/\sigma^2$. In order to to guarantee the whole training process to be (ε,δ) -DP, combining $\alpha(\lambda)$ with Property 2.2, for any $c_0 \in (0,1)$, we choose

$$\begin{aligned}Tq^2s^2\lambda^2/\sigma^2 &= c_0\lambda\varepsilon,\\ \exp((c_0-1)\lambda\varepsilon) &\leq \delta.\end{aligned} \tag{24}$$

Plugging the condition $\lambda \leq \frac{\sigma^2}{s^2}\log\frac{s}{q\sigma}$ into Eq. (24), we derive the bound for ε as $\varepsilon < \frac{1}{c_0}\log\frac{s}{q\sigma}q^2T$ to accomplish (ε,δ)-DP by setting

$$\sigma = \frac{1}{\sqrt{c_0(1-c_0)}}\cdot\frac{qs\sqrt{T\log(1/\delta)}}{\varepsilon}, \tag{25}$$

where $c_0 \in (0,1)$. □

A.3 Privacy Analysis for the Link Reconstruction Based Graph Generation Models with DPSGD

In this section, we conduct the theoretical privacy analysis for link reconstruction based graph generation model based on Lemma 2 and obtain the conclusion of Theorem 1.

Theorem 1. *In training a link reconstruction based graph generation model on a graph with N nodes with batch size as B, given the sampling probability $q = B/N$, and the number of steps T, there exist explicit constants c_1 and c_2 that for any $\varepsilon < c_1 q^2 T$, iteratively updating the model T times with $\tilde{g}_{\theta,\mathcal{L}}$ attains it with (ε, δ)-edge-DP for any $\delta > 0$ if we choose*

$$\sigma \geq c_2 \frac{q\sqrt{T\log(1/\delta)}}{\varepsilon},$$

where $c_1 \geq \frac{1}{c_0}\log\frac{1}{q\sigma}$, $c_2 \leq 1/\sqrt{c_0(1-c_0)}$ for any $c_0 \in (0,1)$.

Proof. Recall the expression of $\tilde{g}_{\theta,\mathcal{L}}$ as

$$\tilde{g}_{\theta,\mathcal{L}} = \frac{1}{N}\left(\sum_{i=1}^{N}\left(\nabla_{v_i,\theta}\mathcal{L}/\max(1, \frac{\|\nabla_{v_i,\theta}\mathcal{L}\|_2}{C})\right) + \mathcal{N}(0, \sigma^2 C^2 \mathbf{I})\right), \tag{26}$$

where $\mathcal{L}$ is the loss function for a link reconstruction based graph generation model, C is the clipping hyper-parameter for the model's original gradient to bound the influence of each node, and σ is the noise scale hyper-parameter. According to the Gaussian Mechanism definition [Dwork *et al.*, 2014]), $\tilde{g}_{\theta,\mathcal{L}}$ is a Gaussian mechanism. Therefore, we first analyze the ℓ_2-norm sensitivity of the clipped gradient function $\tilde{g}_{\theta,\mathcal{L}}$, and then plug the sensitivity value to Lemma 2 and conclude the privacy cost of training DPGVAE, thus finishing the proof for Thereom 1.

Following the graph reconstruction procedure in [Simonovsky and Komodakis, 2018], a single value in the adjacency matrix is sufficient to represent one edge in the respective graph for both directed and undirected graphs. Referring to edge-DP definition [Blocki *et al.*, 2012], though changing an edge in the graph affects 2 nodes for node classification tasks, for a structural inference task, *i.e.*, graph reconstruction, as our work targeting at, adding or removing an edge only results to at most 1 record difference. Together with $\nabla_{v_i,\mathbf{f}}\mathcal{L}$ being clipped as its ℓ_2-norm no more than C, we obtain the sensitivity of $\sum_{i=1}^{N}\nabla_{v_i,\mathbf{f}}\mathcal{L}/\max(1, \frac{\|\nabla_{v_i,\mathbf{f}}\mathcal{L}\|_2}{C})$ as $s = 1 * C = C$.

With plugging in the clipped $\nabla_{v_i,\mathbf{f}}\mathcal{L}$'s sensitivity ($s = C$) into Lemma 2, we derive Theorem 1. We prove that, given the sampling probability $q = B/N$ and the number of steps T, with explicit constants $c_1 \geq \frac{1}{c_0}\log\frac{1}{q\sigma}$ and $c_2 \leq \frac{1}{\sqrt{c_0(1-c_0)}}$, where $c_0 \in (0,1)$, through iteratively updating model T times with Eq. (26), the outcome generation model achieves (ε, δ)-edge-DP for any $\varepsilon < c_1 q^2 T$, and $\delta > 0$ when we choose

$$\sigma \geq c_2 \frac{q\sqrt{T\log(1/\delta)}}{\varepsilon}. \tag{27}$$

□

Specifically, our proof for Theorem 1 serves as a general DP analysis when DPSGD is considered on graph data, which covers our problem as a specific instance appearing to be relatively similar to [Abadi *et al.*, 2016], *i.e.*, obtaining edge-DP for our link reconstruction based graph generation model. However, with a similar analysis of the proof for Theorem 1, we can conduct edge-DP results for other graph models trained with the same technique. For example, for a generative graph model solving the node classification task, changing 1 edge in the input graph affects 2 nodes' representations. Thus, when the gradient value is clipped with C, the sensitivity for the node classification model's clipped gradient function is now 2*C. Moreover, regarding node-DP [Kasiviswanathan *et al.*, 2013] instead of edge-DP, one can also deliver the corresponding sensitivity analysis following our analysis here. For example, for the undirected graph with N nodes without duplicated links nor links starting and ending at the same node, when 1 node in the input graph changes, the respective gradient function clipped with C for training the link reconstruction model shows at most $(N-1)*C$ difference. Therefore, under our Theorem 1 settings, when it comes to node-DP, following the analysis in Theorem 1 with substituting the sensitivity as $(N-1)*C$, node-DP of the mechanism can be similarly concluded.

Recall the training process for the decoder in DPGVAE and the generator in DPGGAN in Section 3. $\mathcal{L}$ in $\tilde{g}_{\theta,\mathcal{L}}$ is substituted with $\mathcal{L}_{rec}$ and $\mathcal{L}_{rec} + \lambda_2\mathcal{L}_{gan}$, respectively. For both $\mathcal{L}_{rec}$ and $\mathcal{L}_{rec} + \lambda_2\mathcal{L}_{gan}$, their gradients are clipped with C and adding Gaussian noises during the training process. Based on Theorem 1, we derive Corollary 1.1 and 1.2 for the decoder in DPGVAE and the generator in DPGGAN respectively as below.

Corollary 1.1 (DPGVAE edge-DP)**.** *Under the same conditions in Theorem 1,iteratively updating the decoder in* DPGVAE *T times with $\tilde{g}_{\theta,\mathcal{L}_{rec}}$ attains it with (ε, δ)-edge-DP.*

Corollary 1.2 (DPGGAN edge-DP)**.** *Under the same conditions in Theorem 1, iteratively updating the generator in* DPGGAN *T times with $\tilde{g}_{\theta,(\mathcal{L}_{rec}-\lambda_2\mathcal{L}_{gan})}$ attains it with (ε, δ)-edge-DP.*

With Corollary 1.1 and 1.2, under specified conditions, the public model (either the decoder in DPGVAE or the generator in DPGGAN) is guaranteed with (ε, δ)-edge-DP by the DP training process. For both DPGVAE decoder and DPGGAN generator updated with noised and clipped representations of the sensitive training graph, they only record noised and partial sensitive information. DPGVAE decoder and DPGGAN generator's link reconstruction procedures, reflecting its training information, only allude to the desensitize information rather than the true sensitive training information. Thus, DPGVAE decoder and DPGGAN generator not only prevent privacy leakage from their inner parameters with DP learning but also preserve the raw private training graphs from being accurately inferred through the respective outputs.

B APPENDIX: More details of experimental results

In this work, we define the goals of secure network release as *preserving global network structure* while *protecting individual link privacy*. In the main content, we have presented

experimental results to support the effectiveness of DPGGAN in both perspectives. That is, for global network structure preservation, we show that the generated graphs of DPGGAN are competitively similar to the original graphs in comparison with the DP-free state-of-the-art graph generative models regarding a suite of commonly concerned global graph statistics. For individual link privacy protection, we show that the links predicted in the generated graphs of DPGGAN are useless (with low accuracy) when evaluated in the original graphs.

The suite of statistics measures the global network structure from different perspectives. As can be inferred from TC, CPL, and GINI, the IMDB networks are in general smaller, tighter, and likely more structurally complex than the DBLP networks, which favors link generation models (*e.g.*, GVAE) over sequence generation models (*e.g.*, NetGAN, and GraphRNN). Consequently, DPGGAN also performs better on the IMDB networks, indicating its advantages in modeling complex link structures as a whole.

In addition to the graph statistics, we further demonstrate the data utility of networks generated by DPGGAN with graph classification, which is the most widely studied graph-level downstream task. We deem this task important towards evaluating network data utility, especially under our consideration of global network structure preservation, because correct graph classification requires the generated graphs to share essential structural properties with the original graphs. As we can see from Table 2 in the main paper, the data utilities evaluated with graph classification are consistent with those evaluated with global graph statistics, as shown in Table 1 in the main paper. Our two DP-constrained models yield highly competitive performance compared with the DP-free state-of-the-art graph generative models.

As for privacy protection, we conduct more detailed inspections of the performance of individual link prediction. In particular, we compare DPGGAN in various privacy budget ε to non-private generative models (GVAE, NetGAN, GraphRNN, and GGAN) to validate how DP protection is reflected on link prediction attacks. We implement this experiment using the node embedding calculated with attri2vec [Zhang *et al.*, 2019]. The results in Figure 3 demonstrate consistency with the rigorous theoretical edge-DP protection in Corollary 1.2, *i.e.*, larger privacy budgets lead to more privacy leakage, allowing attackers to infer individual links in the original networks with higher accuracy.

References

[Abadi *et al.*, 2016] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *SIGSAC*, 2016.

[Blocki *et al.*, 2012] Jeremiah Blocki, Avrim Blum, Anupam Datta, and Or Sheffet. The johnson-lindenstrauss transform itself preserves differential privacy. *FOCS*, 2012.

[Dwork *et al.*, 2014] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Theoretical Computer Science*, 9(3–4):211–407, 2014.

[Kasiviswanathan *et al.*, 2013] Shiva Prasad Kasiviswanathan, Kobbi Nissim, Sofya Raskhodnikova, and Adam D. Smith. Analyzing graphs with node differential privacy. In *TCC*, 2013.

[Simonovsky and Komodakis, 2018] Martin Simonovsky and Nikos Komodakis. Graphvae: Towards generation of small graphs using variational autoencoders. In *ICANN*, 2018.

[Zhang *et al.*, 2019] Daokun Zhang, Jie Yin, Xingquan Zhu, and Chengqi Zhang. Attributed network embedding via subspace discovery. *Data Mining and Knowledge Discovery*, 33(6):1953–1980, 2019.