Scilight

Transactions on Graph Intelligence and Network Applications
https://www.sciltp.com/journals/tgina

Review

Differentially Private Graph Learning: A Survey

Longzhu He [1], Li Sun [1,*], Ming Li [2], Yang Cao [3], Sen Su [1,4,*], Masatoshi Yoshikawa [5], Jia Wu [6], Pietro Liò [7] and Philip S. Yu [8]

[1] School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2] Zhejiang Key Laboratory of Intelligent Education Technology and Application, Zhejiang Normal University, Jinhua 321017, China
[3] Department of Computer Science, Institute of Science Tokyo, Tokyo 145-0061, Japan
[4] School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China
[5] Faculty of Data Science, Osaka Seikei University, Osaka 533-0007, Japan
[6] School of Computing, Macquarie University, Sydney 2113, Australia
[7] Department of Computer Science and Technology, University of Cambridge, Cambridge CB3 0FD, UK
[8] Department of Computer Science, University of Illinois Chicago, Chicago, IL 60607, USA
* Correspondence: lsun@bupt.edu.cn (L.S.); susen@bupt.edu.cn (S.S.)

How To Cite: He, L.; Sun, L.; Li, M.; et al. Differentially Private Graph Learning: A Survey. *Transactions on Graph Intelligence and Network Applications* **2026**.

Received: 31 May 2026
Revised: 8 July 2026
Accepted: 14 July 2026
Published: 30 July 2026

Abstract: Graph learning has been widely applied across diverse domains, including social networks, recommendation systems, and bioinformatics. However, real-world graph data often contains highly sensitive information, raising serious privacy concerns. Differential Privacy (DP) has emerged as a rigorous mathematical framework to protect sensitive information in graph learning while providing formal privacy guarantees. This survey presents the first comprehensive and systematic review of Differentially Private Graph Learning (DPGL). We organize existing DPGL methods into four categories based on the granularity of privacy protection, namely node-level, edge-level, graph-level, and local differential privacy. Within each category, we further analyze learning paradigms, perturbation mechanisms, and their respective strengths and limitations, and identify key technical challenges in the field. Furthermore, we identify future research directions critical for advancing DPGL toward practical deployment in real-world applications. This survey aims to provide a unified reference for researchers and practitioners while inspiring future innovations in privacy-preserving graph learning.

Keywords: differential privacy; graph learning; privacy-preserving

1. Introduction

Graph learning has demonstrated remarkable success across various domains, such as social networks [1,2], recommendation systems [3], anomaly detection [4], and bioinformatics [5]. In these areas, graph-structured data is typically used to make predictions about nodes, edges, or entire graphs. While many graph learning methods, such as Graph Neural Networks (GNNs) [6], have proven to be highly effective, real-world graph data often contains sensitive personal information, such as user profiles, social connections, and behavioral patterns, posing significant privacy risks. Training graph learning models on such data can expose sensitive information to malicious attackers, who may be able to infer private details by interacting with the model [7–9]. Therefore, it is crucial to develop privacy-preserving graph learning models to safeguard data privacy.

Differential Privacy (DP) [10] is regarded as the gold standard for providing formal privacy guarantees for algorithms processing sensitive data. It introduces calibrated noise to limit the influence of any single data item, preventing attackers with partial data from inferring sensitive information about a specific individual. DP offers a strong theoretical guarantee, making it one of the most reliable privacy-preserving mechanisms. Its widespread application has made it a standard in fields such as machine learning and data mining [11,12].

Differentially Private Graph Learning (DPGL) has recently emerged as an effective solution to address privacy concerns in graph learning. By incorporating calibrated DP noise into the graph learning process, DPGL protects sensitive information in graph data while preserving model utility. As shown in Figure 1, two main scenarios for

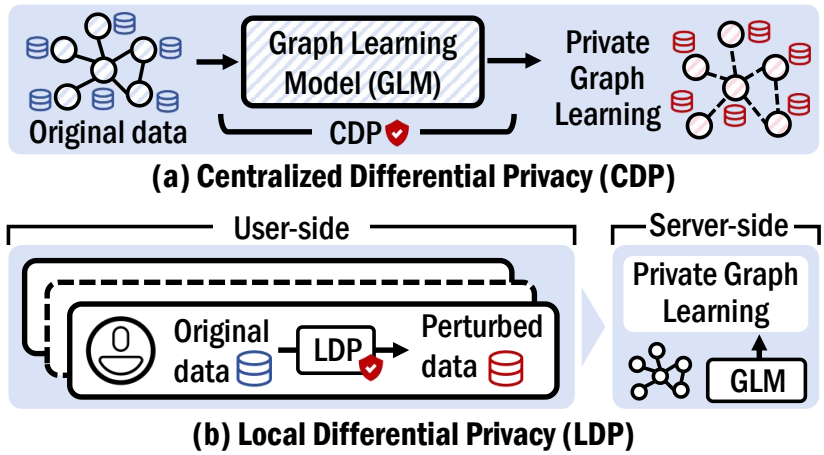

Figure 1. Two main differentially private graph learning scenarios. (a) CDP protects graph data by adding noise at different levels of granularity (e.g., node, edge, or graph level), ensuring that privacy is maintained across the entire dataset. (b) LDP offers a stricter privacy guarantee by adding noise directly to decentralized user data before server-side graph learning, ensuring privacy at the user level.

DP in graph learning are considered: Centralized Differential Privacy (CDP) [10], depicted in Figure 1a, and Local Differential Privacy (LDP) [13], shown in Figure 1b. Within the CDP setting, three privacy granularities are commonly considered: ① node-level DP [14], ② edge-level DP [15], and ③ graph-level DP [5]. Specifically, node-level DP protects individual nodes by adding noise to node-specific data, edge-level DP ensures privacy at the edge level by perturbing the connections between nodes, and graph-level DP protects the entire graph by preventing the leakage of structural information. In contrast, LDP, as illustrated in Figure 1b, enforces privacy protection at the user level by perturbing decentralized user data before server-side graph learning. Depending on the privacy paradigm and the granularity at which noise is injected, these settings provide different trade-offs between privacy protection and learning utility.

Despite the rapid growth of DPGL in recent years, no comprehensive survey dedicated to this field currently exists. While existing surveys [16,17] have examined privacy-preserving techniques for graph-structured data, they largely predate the recent surge in DPGL research and do not systematically study DP in graph learning settings. This survey fills this critical gap. We review existing works under different DP variants, including node-level, edge-level, graph-level, and local DP. Additionally, we highlight studies that go beyond privacy and examine DPGL from a broader trustworthy graph learning perspective [18], as privacy protection alone is insufficient to guarantee reliable graph learning in practical deployments. Figure 2 provides an overview of the taxonomy of DPGL approaches, and Table 1 presents a systematic summary of existing methods and their key characteristics to facilitate comparison. We also maintain a GitHub repository https://github.com/thisxyz/Awesome-DPGL (accessed on 20 January 2026) to provide up-to-date resources. Finally, we discuss the challenges and future research directions in this rapidly evolving field, highlighting promising areas for further investigation. Overall, this survey aims to serve as a valuable reference for researchers and practitioners working on privacy-preserving graph learning by providing a comprehensive overview of recent advances in DPGL.

2. Preliminaries

In this section, we begin by presenting the problem definition (Section 2.1), followed by background on differential privacy (Section 2.2) and graph learning models (Section 2.3), and conclude with a taxonomy of DPGL methods (Section 2.4).

2.1. Problem Definition

Consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X}, \mathbf{Y})$, where $\mathcal{V}$ and $\mathcal{E}$ denote the sets of nodes and edges, respectively, $\mathbf{X}$ is the node feature matrix, and $\mathbf{Y}$ contains supervisory information such as node- or graph-level labels. The goal of graph learning is to learn effective representations from graph-structured data to support downstream tasks, such as node classification [6]. In real-world scenarios, graph data often contain sensitive information (e.g., user profiles and social connections), which makes graph learning models vulnerable to privacy leakage during training or deployment. DPGL aims to design graph learning algorithms that provide rigorous privacy guarantees for sensitive information while maintaining model utility (e.g., node classification accuracy).

Table 1. Summary and taxonomy of DPGL approaches. NDP, EDP, and GDP denote node-level, edge-level, and graph-level DP, respectively. Meth. denotes the DP mechanism employed in each method, including the Gaussian Mechanism (GM), Laplace Mechanism (LAP), Randomized Response (RR), 1-Bit (1B), Multi-Bit (MB), Piecewise Mechanism (PM), Exponential Mechanism (EM), Square Wave (SW), and Optimal Unary Encoding (OUE). Task refers to the downstream tasks, such as node classification (NC), link prediction (LP), graph classification (GC), graph clustering (GCI), node clustering (NCI), sequential recommendation (SR), influence maximization (IM), structural equivalence (SE), and edge sign prediction (ESP). Main Metric denotes the primary evaluation metric, including Accuracy (ACC), F1 score, Area Under the ROC Curve (AUC), Sensitivity, Specificity, Normalized Discounted Cumulative Gain at rank K (NDCG@K), Precision, Recall at rank K (Recall@K), Mean Reciprocal Rank at rank K (MRR@K), Influence Spread (IS), Coverage Ratio (CR), StrucEqu, Mutual Information (MI), Topology Privacy Leakage (TPL), and Symmetric Separation Index (SSI). ✓ and ✗ indicate whether a method supports the corresponding property.

Method	NDP	EDP	GDP	LDP	Meth.	ϵ	Model	Task	Main Metric	Publication	Reference
LPGNN	✗	✗	✗	✓	MB, RR	[0.01, 3.00]	GCN, GAT, SAGE	NC	ACC	CCS	[19]
DP-GNN	✓	✗	✗	✗	GM	[1.00, 20.00]	GCN, GAT, GIN, MLP	NC	ACC, F1 score	PAIR2Struct	[20]
LapGraph	✗	✓	✗	✗	LAP	[1.00, 10.00]	GCN	NC	F1 score, AUC	S&P	[7]
DPGNN	✗	✗	✓	✗	GM	[0.50, 20.00]	GCN, GAT, SAGE, Cheb	GC	ACC, AUC, Sensitivity, Specificity, F1 score	TPAMI	[5]
Solitude	✗	✗	✗	✓	MB, RR	[0.50, 8.90]	GCN, SAGE	NC	ACC	TIFS	[21]
LPGNet	✗	✓	✗	✗	LAP	[1.00, 10.00]	MLP	NC	F1 score, AUC	CCS	[15]
GW	✗	✗	✗	✓	MB	[0.01, 10.00]	SVM	GC, GCI	ACC	IJCAI	[22]
PrivGNN	✓	✗	✗	✗	LAP	[2.83, 170.41]	SAGE	NC	ACC	TMLR	[23]
Blink	✗	✗	✗	✓	LAP, RR	[1.00, 8.00]	GCN, GAT, SAGE	NC	ACC	CCS	[1]
GAP	✓	✓	✗	✗	GM	[0.10, 16.00]	MLP	NC	ACC, AUC	Security	[24]
LGA-PGNN	✗	✗	✗	✓	PM	[0.01, 20.00]	SAGE	NC	ACC	TIFS	[25]
LPL-LPF-IFGNN	✗	✗	✗	✓	MB, RR	[0.02, 2.00]	GCN, SGC	NC	ACC, NDCG@K	KBS	[26]
DPDGC	✓	✓	✗	✗	GM	[1.00, 16.00]	MLP	NC	ACC	NeurIPS	[27]
HeterPoisson	✓	✗	✗	✗	GM	[2.00, 16.00]	GCN, GIN, SAGE	NC	ACC, Precision	S&P	[14]
ProGAP	✓	✓	✗	✗	GM	[0.25, 32.00]	MLP, JK	NC	ACC	WSDM	[28]
Eclipse	✗	✓	✗	✗	GM, LAP	[0.10, 8.00]	GCN	NC	F1 score, AUC	PETS	[29]
DPAR	✓	✗	✗	✗	GM, EM	[1.00, 8.00]	APPR	NC	ACC	WWW	[30]
LinkGuard	✗	✗	✗	✓	LAP, RR	[1.00, 4.00]	GCN, GAT	NC	ACC	WWW	[31]
DPLP	✗	✓	✗	✗	GM	[1.00, 10.00]	GCN, SAGE, GIN, DGCNN	LP	AUC	ICDE	[32]
DIPSGNN	✗	✗	✗	✓	PM, OUE, GM	[3.00, 30.00]	GGNN	SR	Recall@K, MRR@K	TKDD	[3]
SL-LPGL	✗	✗	✗	✓	LAP	[0.02, 2.00]	MLP, GCN	NC	ACC	KBS	[33]
DPRR	✗	✗	✗	✓	LAP, RR	[0.20, 2.00]	GIN	GC	ACC, AUC	TDP	[34]
RGNN	✗	✗	✗	✓	RR	[0.01, 3.00]	GCN, GAT, SAGE	NC	ACC	SDM	[35]
PrivGE	✗	✗	✗	✓	SW	[0.01, 5.00]	MLP, LR	NC, LP	ACC, AUC	CIKM	[36]
GraphPrivatizer	✗	✗	✗	✓	MB, RR	[0.10, 8.00]	GCN, GAT, SAGE, GATv2, GConv, GT	NC	ACC, AUC	TMLR	[37]
TDP-GNN	✗	✗	✗	✓	LAP	[5.00, 15.00]	GCN, SAGE	NC	ACC	WWW	[38]
PrivIM	✓	✗	✗	✗	GM	[1.00, 6.00]	GCN, GAT, SAGE, GIN	IM	IS, CR	ICDE	[39]
SE-PrivGEmb	✓	✗	✗	✗	GM	[0.50, 3.50]	SGM	LP, SE	AUC, StrucEqu	ICDE	[40]
AdvSGM	✓	✗	✗	✗	GM	[1.00, 6.00]	SGM	LP, NCI	AUC, MI	ICDE	[41]
GCON	✗	✓	✗	✗	LAP	[0.50, 4.00]	GCN	NC	F1 score	ICDE	[42]
DPGNN	✗	✗	✓	✗	GM	[1.00, 12.00]	GCN, GAT, SAGE	GC	ACC	ESWA	[43]
DP-kNN	✗	✗	✓	✗	GM	[3.00, 7.00]	GCN, MLP, kNN	GC	ACC	ESWA	[44]
GRASP	✗	✓	✗	✗	GM	[0.13, 2.00]	GCN, GAT	NC	ACC, AUC	SIGKDD	[45]
DP-PGR	✗	✓	✗	✗	GM, LAP	[1.00, 9.00]	GCN, GAT, SAGE	NC	ACC, TPL	CCS	[46]
ADP-GNN	✓	✗	✗	✗	GM	[2.00, 16.00]	MLP	NC	ACC	COMNET	[47]
HPGR	✗	✗	✗	✓	LAP, RR	[1.00, 8.00]	GCN, GAT, SAGE	NC	ACC	ECML-PKDD	[48]
UPGNet	✗	✗	✗	✓	MB, PM	[0.01, 3.00]	GCN, GAT, SAGE	NC	ACC	ICML	[49]
DPRO-GNN	✗	✗	✓	✗	GM	[0.50, 20.0]	MLP	GC	ACC, AUC, Sensitivity, Specificity, F1 score	INS	[50]
ASGL	✓	✗	✗	✗	GM	[1.00, 6.00]	LR	ESP, NCI	AUC, SSI	SIGKDD	[51]
AttackDPGL	✗	✗	✗	✓	MB, PM, SW	[0.01, 1.00]	GCN, GAT, SAGE	NC, LP	ACC, AUC	SIGKDD	[52]
CureNet	✗	✗	✗	✓	1B, GM, LAP, MB, PM, SW	[0.01, 10.00]	GCN, GAT, SAGE, APPNP, SGC, SSGC	NC, LP	ACC, AUC	WWW	[53]
CARIBOU	✓	✓	✗	✗	GM	[1.00, 32.00]	CGL, MLP	NC	ACC, AUC	NDSS	[54]

2.2. Differential Privacy (DP)

DP [10] provides rigorous privacy guarantees by limiting the impact of any single data record by noise injection, and has been widely adopted in privacy-preserving data analysis and machine learning [12,55]. Formally, DP is defined as follows:

Definition 1. *(ϵ, δ)-DP). Let $\mathcal{D}$ and $\mathcal{D}'$ be two neighboring datasets that differ in at most one entry. A randomized mechanism $\mathcal{M}$ satisfies (ϵ, δ)-DP if for all $\mathcal{S} \subseteq Range(\mathcal{M})$:*

$$Pr\left[\mathcal{M}(\mathcal{D}) \in \mathcal{S}\right] \leq e^{\epsilon} Pr\left[\mathcal{M}(\mathcal{D}') \in \mathcal{S}\right] + \delta, \tag{1}$$

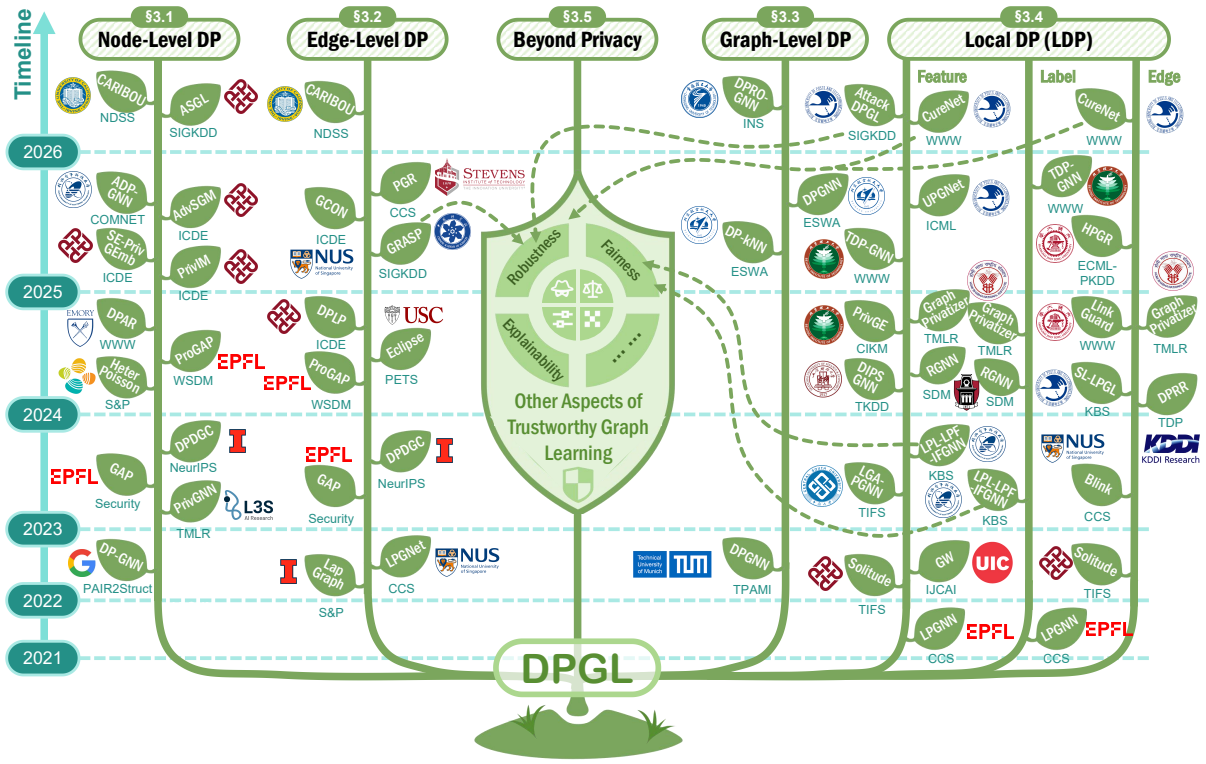

Figure 2. Taxonomy of differentially private graph learning (DPGL) approaches. DPGL is categorized into node-level DP (Section 3.1), edge-level DP (Section 3.2), graph-level DP (Section 3.3), and LDP (Section 3.4), with an additional category beyond privacy in trustworthy graph learning (Section 3.5).

where $\mathcal{M}(\mathcal{D})$ represents the output of $\mathcal{M}$ with the input $\mathcal{D}$, ϵ and δ are the privacy parameters, with ϵ often referred to as the privacy budget; smaller values indicate stronger privacy.

In graph learning, the notion of neighboring datasets varies depending on the privacy granularity, leading to several DP variants.

- Node-level DP [14] treats an individual node together with all its incident edges as a single privacy unit, aiming to protect whether a user participates in the graph. This setting is particularly suitable for social networks, recommendation systems, and citation networks, where revealing the existence of a user or entity itself may disclose sensitive information.
- Edge-level DP [15] considers two graphs to be neighboring if they differ by a single edge, providing weaker but more utility-friendly guarantees. It is well suited for scenarios where the relationship between entities is sensitive, such as friendship links, communication records, or transaction connections, while the entities themselves can be publicly known.
- Graph-level DP [5] extends privacy protection to entire graphs and is commonly used in collections of graphs, such as molecular property prediction, brain network analysis, and biological graph classification, where each graph corresponds to a sensitive individual sample.
- In contrast to these centralized settings, Local Differential Privacy (LDP) [13] perturbs data locally at the user side before sharing, eliminating the need for a trusted curator at the cost of higher noise and reduced utility. LDP is especially attractive in decentralized and large-scale platforms, such as mobile applications, online social networks, and telemetry collection systems, where users may be unwilling to share raw graph data with a central server.

2.3. Graph Learning Models

Graph learning models aim to learn expressive representations from graph data by jointly exploiting node features and graph topology. Among various graph learning paradigms, GNNs [56,57] have emerged as the dominant class of models, as they can effectively capture both structural and feature information of nodes and graphs. Most

GNNs follow a message-passing framework, where each node iteratively aggregates information from its neighbors and updates its representation. Formally, at the l-th layer, the aggregation and update steps for node v are given by:

$$\mathbf{M}_v^l = \textsc{Aggregate}(\mathbf{H}_u^{l-1} : u \in \mathcal{N}_v), \tag{2}$$

$$\mathbf{H}_v^l = \textsc{Update}(\mathbf{H}_v^{l-1}, \mathbf{M}_v^l), \tag{3}$$

where $\mathbf{M}_v^l$ denotes the aggregated neighborhood message for node v at layer l, $\mathbf{H}_v^l$ denotes the representation of node v at layer l, $\mathcal{N}_v$ is the set of neighbors of v, and AGGREGATE and UPDATE are differentiable functions (e.g., mean, sum, or neural network layers) that learn to combine neighborhood information. This framework underlies most modern GNN architectures, e.g., GCN, GraphSAGE, GAT, GIN, SGC, and APPNP.

2.4. Taxonomy

In this survey, we organize DPGL methods along two complementary dimensions: the trust model and the granularity of privacy protection. Under the centralized DP setting, methods are categorized according to the protected graph object into node-level, edge-level, and graph-level DP. In contrast, LDP represents a distinct trust model, where data are perturbed at the source before collection. Since LDP can be applied to different graph components, we further organize LDP methods according to the protected information, including feature, edge, and label privacy. Other research directions, such as federated graph learning with DP, also study private graph analysis but introduce additional assumptions and system settings, such as distributed data ownership or cross-device training, which are not aligned with the focus of this survey. Likewise, DP variants that fall outside the centralized or local graph learning paradigms are not considered. Consequently, our taxonomy provides a clear and consistent framework for organizing and comparing existing DPGL methods under standard graph learning settings.

3. Differentially Private Graph Learning

In this section, we sequentially introduce DPGL under node-level DP (Section 3.1), edge-level DP (Section 3.2), graph-level DP (Section 3.3), and LDP (Section 3.4) settings. In Section 3.5, we further discuss aspects beyond privacy, such as robustness, fairness, and other trustworthiness considerations.

3.1. DPGL under Node-Level DP

In the node-level DP setting, the objective is to protect each individual node together with all its associated edges, while maintaining satisfactory graph learning performance. Compared with edge-level DP, node-level DP entails substantially higher sensitivity, since the removal of a single node simultaneously alters all adjacent edges. Moreover, under multi-hop message passing in GNNs, the influence of a node can propagate far beyond its immediate neighborhood, potentially affecting the entire graph through long-range dependency propagation. To this end, existing approaches can be broadly categorized into four sensitivity control strategies: ① Sampling-based Bounding, ② Decoupling Topology from Features, ③ Knowledge Distillation, and ④ Adversarial Training.

3.1.1. Sampling-Based Bounding

These methods limit the influence scope of each node through sampling techniques, transforming unbounded node sensitivity into bounded quantities to effectively control noise magnitude. For example, DP-GNN [20] enforces an upper bound on the number of neighbors per node via subgraph sampling. HeterPoisson [14] introduces heterogeneous Poisson sampling, where neighbor nodes are assigned sampling probabilities inversely proportional to their out-degrees, thereby balancing the impact of high-degree nodes. PrivIM [39] proposes a dual-stage adaptive frequency sampling scheme for Influence Maximization (IM): the first stage dynamically adjusts sampling probabilities to mitigate inter-node dependencies, while the second stage supplements boundary subgraphs to enrich structural information without consuming additional privacy budget, resulting in the first node-level DP solution for IM.

3.1.2. Decoupling Topology from Features

Decoupled learning architecture separates topology aggregation from feature learning, avoiding repeated noise injection through pre-aggregation and caching, thereby decoupling the privacy budget from subsequent model training. GAP [24] adopts a three-stage framework that privately aggregates k-hop neighborhood information once and caches the noisy results, enabling downstream feature learning without additional privacy cost. ProGAP [28] extends GAP via progressive training to capture multi-scale neighborhood information, while ADP-GNN [47]

introduces an adaptive differentially private noise mechanism that dynamically adjusts the noise magnitude at each iteration based on historical and current model parameters, improving the privacy-utility trade-off. DPDGC [27] introduces fine-grained decoupling by designing distinct privacy mechanisms for structural propagation and feature aggregation, minimizing the impact of structural noise on feature learning. DPAR [30] preprocesses graph structure using differentially private approximate personalized PageRank (DP-APPR), decoupling structural sensitivity from gradient updates and enabling deeper GNNs without cumulative noise. While these approaches demonstrate strong empirical performance, their privacy budget accumulation in deep decoupled architectures was not fully understood. CARIBOU [54] addresses this limitation by establishing sublinear privacy budget growth under certain decoupled designs, providing rigorous guarantees for constructing deeper DPGL models.

3.1.3. Knowledge Distillation

These methods leverage teacher-student frameworks to distill knowledge from private graphs into models trained on public graphs, thereby avoiding direct access to sensitive data. PrivGNN [23] instantiates this idea using the PATE framework [58], where private nodes are partitioned into disjoint subsets to train multiple teacher models. For public query nodes, teachers cast noisy votes via Laplace-perturbed aggregation, and a student model is subsequently trained on public data. This paradigm is particularly suitable for settings with auxiliary public graphs.

3.1.4. Adversarial Training

Adversarial methods leverage the natural robustness of adversarial learning to noisy interactions, embedding noise injection into the optimization process rather than simple additive Gaussian noise. AdvSGM [41] designs two optimizable noise terms corresponding to skip-gram parameters, achieving DP gradient updates through adversarial module weight tuning without additional noise injection, significantly outperforming existing methods in link prediction and node clustering. SE-PrivGEmb [40] designs a unified noise tolerance mechanism for structure-preference settings by perturbing non-zero vectors to accommodate arbitrary structure preferences, theoretically proving the preservation of arbitrary node proximities under node-level DP. ASGL [51] decomposes signed graphs into positive and negative subgraphs and employs gradient-perturbed adversarial modules to approximate the signed connectivity distributions for more accurate learning. With balance theory and constrained BFS, it reduces error propagation from edge perturbations while preserving model utility and robustness.

Takeaway: Under the node-level DP setting, protecting each node and its incident edges results in high sensitivity due to recursive neighborhood aggregation. Existing solutions focus on four complementary strategies: ① bounding sensitivity via sampling, ② decoupling topology from features to reduce cumulative noise, ③ distilling knowledge from private to public graphs, and ④ integrating adversarial training to enhance robustness to noise. These approaches reveal a common design principle: rather than relying solely on stronger perturbation mechanisms, they seek to reduce the impact of privacy noise through sensitivity reduction, noise isolation, or knowledge transfer. Consequently, recent research has gradually shifted from directly injecting noise to designing privacy-aware graph learning architectures that better balance privacy, utility, and scalability.

3.2. DPGL under Edge-Level DP

In the edge-level DP setting, the goal is to protect the presence of individual edges while preserving graph learning utility. Existing edge-level DP methods can be categorized into three paradigms: ① Graph Perturbation, ② Decoupling Topology from Features, and ③ Generative Defense.

3.2.1. Graph Perturbation

These methods inject calibrated noise into raw graph data, intermediate representations, or training objectives, where the key challenge is to balance privacy and utility through effective sensitivity control. LapGraph [7] perturbs the adjacency matrix using the Laplace mechanism, but direct structural perturbation severely distorts graph topology and leads to notable utility loss. LPGNet [15] reduces noise by encoding graphs with low-dimensional cluster degree vectors before perturbation, at the cost of discarding fine-grained connectivity information. Eclipse [29] exploits the low-rank structure of graphs and injects noise only into singular values, substantially reducing perturbation dimensionality while preserving primary topology. GCON [42] perturbs the training objective instead of the graph structure, achieving edge-level privacy while maintaining the original message-passing process. GRASP [45] targets graph reconstruction attacks by introducing noise via a Bernoulli-based mechanism to disrupt embedding similarity and relative rankings.

3.2.2. Decoupling Topology from Features

Decoupled learning architecture designs aim to reduce privacy budget consumption by explicitly separating sensitive from non-sensitive computations in graph learning pipelines. The core idea is to preprocess graph structure and cache intermediate aggregation results, thereby avoiding repeated noise injection across multiple GNN layers and substantially mitigating privacy budget accumulation. GAP [24], ProGAP [28], DPDGC [27], and CARIBOU [54] support both node-level DP and edge-level DP. As described in Section 3.1, these methods achieve privacy protection through decoupling topology aggregation from feature learning, progressive training strategies, multi-granular topology protection, and sublinear privacy budget growth. They maintain consistent architectural principles for edge-level DP, effectively reducing data dependency through pre-aggregation and caching techniques. DPLP [32] proposes a path-based subgraph extraction strategy for link prediction, retaining only paths between target node pairs to reduce sensitivity and inter-subgraph dependency. By extracting k-hop path subgraphs, it preserves global heuristic information and enables shallow GNNs to achieve expressive power comparable to deeper models. Experiments demonstrate that DPLP consistently outperforms direct perturbation methods.

3.2.3. Generative Defense

Unlike passive noise injection, generative defense methods proactively learn graph distributions and generate privacy-preserving synthetic data for training. DP-PGR [46] targets topology inference attacks, which existing edge-level DP methods either fail to defend against or address at the cost of significant utility degradation. It formulates private graph reconstruction as a bi-level optimization problem and reconstructs synthetic graphs that jointly account for model utility and topology privacy, thereby substantially reducing topology leakage while preserving model accuracy.

Takeaway: Edge-level DP aims to conceal the existence of individual edges with lower sensitivity than node-level DP, yet remains vulnerable to utility degradation caused by structural perturbations. Existing approaches can be grouped into three paradigms: ① direct graph or objective perturbation to enforce privacy, ② decoupled architectures that mitigate cumulative noise through topology pre-aggregation and caching, and ③ generative defenses that proactively reconstruct privacy-preserving synthetic graphs. Collectively, these methods reflect a shift from naive structural noise injection toward structured and distribution-aware designs that better preserve connectivity patterns under strict edge-level privacy guarantees. This evolution reveals a common design principle: the primary challenge in edge-level DP lies not only in enforcing privacy through edge perturbation, but also in preserving structural information essential for message passing. Consequently, recent methods increasingly integrate topology-aware optimization, graph reconstruction, and structural priors into the learning process to compensate for privacy-induced distortions and achieve a better privacy–utility trade-off.

3.3. DPGL under Graph-Level DP

Graph-level DP protects the participation of an entire graph by treating each graph as an indivisible privacy unit. This setting naturally arises in whole-graph learning tasks such as molecular and biological graph classification, where each graph represents a sensitive entity. Compared with node-level and edge-level DP, graph-level DP operates at a coarser privacy granularity and imposes tighter learning constraints, often leading to substantial utility degradation. As a result, DPGL under graph-level DP remains relatively underexplored. Existing approaches primarily build upon DP-SGD by applying gradient clipping and noise injection at the graph level during GNN training. Early work by Mueller et al. [5] establishes this paradigm and demonstrates its feasibility for graph-level privacy protection. Subsequent studies seek to mitigate the severe utility loss induced by strong privacy constraints through complementary design choices. For instance, some methods enhance representation quality via contrastive or semi-supervised learning [44], while others focus on improving optimization stability by adapting privacy budgets, tuning clipping strategies, or introducing adaptive DP optimizers [43]. More recent work further explores hierarchical pooling and bias-corrected DP optimizers to stabilize training under graph-level noise [50]. Despite these efforts, achieving a favorable trade-off between strong graph-level privacy guarantees and high model utility remains challenging.

Takeaway: Graph-level DP protects the participation of entire graphs in a dataset, making it particularly suitable for graph classification and graph-level learning tasks. Existing DPGL methods under graph-level DP are limited in number and primarily rely on gradient perturbation and optimization-level techniques to mitigate privacy noise. This reflects a fundamental challenge: graph-level perturbation affects holistic graph representations, making utility recovery significantly more difficult. Consequently, existing methods mainly focus on improving optimization efficiency and reducing optimization-induced noise, rather than exploiting graph structural priors.

Table 2. Overview of protection targets and perturbation mechanisms in LDP-based graph learning. ✓ denotes that the corresponding data type is protected, while ✗ indicates that it is not considered. Meth. denotes the DP mechanism employed in each method.

Method	Feature	Meth.	Edge	Meth.	Label	Meth.
LPGNN [19]	✓	MB	✗	-	✓	RR
Solitude [21]	✓	MB	✓	RR	✗	-
GW [22]	✓	MB	✗	-	✗	-
Blink [1]	✗	-	✓	LAP, RR	✗	-
LGA-PGNN [25]	✓	PM	✗	-	✗	-
LPL-LPF-IFGNN [26]	✓	MB	✗	-	✓	RR
SL-LPGL [33]	✗	-	✓	LAP	✗	-
DPRR [34]	✗	-	✓	LAP, RR	✗	-
DIPSGNN [3]	✓	PM, OUE	✓	GM	✗	-
LinkGuard [31]	✗	-	✓	LAP, RR	✗	-
RGNN [35]	✓	RR	✗	-	✓	RR
PrivGE [36]	✓	SW	✗	-	✗	-
GraphPrivatizer [37]	✓	MB	✓	RR	✓	RR
TDP-GNN [38]	✓	LAP	✗	-	✓	RR
HPGR [48]	✗	-	✓	LAP, RR	✗	-
UPGNet [49]	✓	MB, PM	✗	-	✗	-

Developing structure-aware graph-level privacy mechanisms that better preserve graph semantics while maintaining rigorous privacy guarantees remains an important direction for future research.

3.4. DPGL under LDP

In the LDP setting, users perform privacy protection locally, and only perturbed graph data is made available to the server. This strict trust model substantially reduces the risk of privacy leakage, since raw data is never shared with the server. However, the strong noise injected at the data source also poses significant challenges to the utility of graph learning models. From the perspective of what information is protected, existing studies on LDP-based graph learning can be broadly categorized into three complementary directions, namely ① Feature Privacy, ② Edge Privacy, and ③ Label Privacy. A summary of the protected objects and the corresponding perturbation mechanisms adopted in existing LDP-based methods is provided in Table 2. In the following, we systematically review representative methods along these three directions.

3.4.1. Feature Privacy

The feature privacy scenario aims to protect node features by requiring each node to locally perturb its feature vector before interacting with an untrusted server. However, as node feature vectors are typically high-dimensional, naively applying conventional noise mechanisms independently to each dimension leads to severe noise accumulation, substantially degrading the utility of graph learning tasks. To address this challenge, existing studies primarily follow two research lines.

One line of research focuses on developing more effective LDP perturbation mechanisms to reduce noise while preserving privacy guarantees. LPGNN [19] introduces the Multi-Bit (MB) mechanism, which extends the classic one-bit mechanism [59] to jointly perturb multi-dimensional node features. Subsequent methods, including Solitude [21], GW [22], and GraphPrivatizer [37], adopt MB as a core primitive for node feature protection and further extend it to diverse application scenarios. RGNN [35] applies Randomized Response (RR) [60] mechanism to both node features and labels, and investigates how to reconstruct useful information from perturbed data. LGA-PGNN [25] proposes a privacy-preserving GNN framework based on local graph augmentation, where node features are perturbed using the Piecewise Mechanism (PM) [61]. DIPSGNN [3] further extends LDP-based perturbation to sequential recommendation systems by leveraging both PM and RR. PrivGE [36] designs a novel LDP perturbation mechanism based on the Square LongzhuWave (SW) [62] mechanism to obfuscate node feature vectors, achieving superior performance on downstream tasks.

Another research line focuses on better calibrating and exploiting noisy features through the optimization of the graph learning pipeline. UPGNet [49] explicitly considers two key factors affecting the utility of locally private graph learning, including feature dimension d and neighborhood size $|\mathcal{N}(v)|$, which significantly enhances graph

learning utility under strong privacy guarantees. TDP-GNN [38] proposes a topology-aware differentially private GNN framework to enable personalized privacy protection.

3.4.2. Edge Privacy

In the edge privacy scenario, randomized response [60] is widely adopted to protect graph structures. Building upon this mechanism, a line of research explores how to mitigate the utility degradation caused by structural perturbations from multiple perspectives, including regularization, Bayesian estimation, degree-aware protection, and graph structure learning. Solitude [21] is among the earliest works to extend LDP to graph structures by collecting local structural views via RR, and further applies sparse regularization during GNN training to suppress the impact of injected noise. Blink [1] and LinkGuard [31] improve utility through Bayesian denoising and dynamic server-side topology refinement during training, respectively. DPRR [34] preserves node degree information by combining RR with strategically sampled edges, where sampling probabilities are calibrated leveraging the Laplace mechanism. GraphPrivatizer [37] maintains topological consistency via structure-aware neighbor replacement, while HPGR [48] reconstructs perturbed edges by leveraging graph homophily priors. Finally, SL-LPGL [33] adopts a graph structure learning paradigm to recover high-quality graphs from low-dimensional structural encodings.

3.4.3. Label Privacy

In semi-supervised learning, a subset of nodes is associated with ground-truth labels, which may contain sensitive semantics, such as disease types or credit ratings. To mitigate the risk of label privacy leakage, these labels are required to be locally perturbed before being used for model training. Unlike high-dimensional node features or complex graph structures, labels are typically low-dimensional discrete variables but exhibit higher semantic sensitivity. Existing studies [37, 38] mainly rely on RR mechanisms to perturb labels. To mitigate the negative impact of noisy labels on graph learning utility, LPGNN [19] introduces a Drop training strategy to reduce noise propagation. RGNN [35] also applies RR-based perturbation, but reconstructs the label distribution via frequency estimation, achieving better performance in label-scarce settings.

Takeaway: LDP enforces privacy at the data source, at the cost of substantial noise. Existing methods can be grouped into three complementary directions: ① Feature privacy tackles high-dimensional noise through improved local perturbation mechanisms and noise-aware learning strategies. ② Edge privacy mitigates structural distortion via denoising, reconstruction, and structure learning techniques. ③ Label privacy typically relies on randomized response, but remains relatively underexplored. These approaches share a common principle: rather than relying solely on stronger perturbation, they aim to recover useful information and improve robustness against privacy noise. Consequently, recent LDP graph learning research has shifted toward privacy-aware architectures that better balance privacy, utility, and scalability.

3.5. Beyond Privacy

While DP provides rigorous privacy guarantees for graph learning, privacy protection represents only one dimension of building trustworthy graph learning systems [18]. In practical deployments, DPGL models must also satisfy other critical trustworthiness requirements, such as robustness and fairness. However, DP mechanisms may interact with or even conflict with these objectives: differential noise injection can amplify data biases, DP perturbations may introduce vulnerabilities to poisoning or reconstruction attacks, and privacy budget allocation may unevenly affect different sensitive groups. Consequently, simultaneously achieving robustness, fairness, and other trustworthiness objectives beyond privacy remains a key challenge for deploying DPGL in real-world applications. For instance, regarding robustness, AttackDPGL [52] reveals that LDP protocols are vulnerable to data poisoning, where adversaries exploit privacy noise to inject malicious updates and degrade downstream GNN performance. Subsequent work [53] introduces a defense framework CureNet that detects malicious participants and restores graph utility under LDP, substantially improving the robustness of locally private graph learning. From the defense perspective, GRASP [45] mitigates graph reconstruction attacks via a structured perturbation mechanism that regulates embedding similarity shifts. Regarding fairness, LPF-IFGNN [26] studies individual fairness under LDP and shows that privacy noise disproportionately harms marginalized nodes, exacerbating biased predictions. It further proposes normalization-aware perturbation and multi-hop denoising to jointly improve privacy and fairness.

Takeaway: Recently, a few studies have explored the interactions between privacy and other trustworthiness dimensions, such as robustness and fairness. However, joint optimization across multiple trustworthiness objectives remains largely underexplored, and other dimensions, for example explainability and well-being, have received little attention. Beyond ensuring privacy protection, simultaneously investigating and achieving robustness, fairness, and other trustworthiness goals constitutes a critical challenge for DPGL toward real-world deployment.

4. Challenges and Future Directions

4.1. Privacy-Utility Trade-off under Strong Privacy

Balancing privacy protection and model utility remains one of the most fundamental challenges in DPGL. Existing approaches (e.g., GAP [24]) generally exhibit stable performance only under relatively relaxed privacy budgets, while model utility often degrades sharply when stronger privacy constraints are imposed. Due to the inherent and complex structural dependencies in graph data, protecting a single node or edge may affect information across a large neighborhood, making this challenge more pronounced than in non-structured data domains. Future research should explore more adaptive and structure-aware noise injection and privacy budget allocation strategies that better exploit graph properties, in order to maximize utility under strict privacy guarantees.

4.2. Scalability to Large-Scale Graphs

Scalability constitutes a key bottleneck for bringing DPGL into real-world applications. Repeated neighborhood sampling, gradient clipping, and privacy accounting introduce substantial computational and memory overhead, limiting the applicability of existing methods to large-scale graphs. Promising future directions include the design of efficient private mini-batch training strategies, structure-aware graph sampling methods, and more efficient distributed DP graph learning frameworks to support training on graphs with millions or even billions of nodes.

4.3. DP for Graph Foundation Models (GFMs)

GFMs [63] offer promising transferability and generalization for privacy-preserving graph learning, yet remain largely unexplored under DP. Most existing models are pre-trained on non-private data, limiting their applicability in sensitive scenarios. The key challenge is maintaining transferable representations under DP constraints, as standard DP-SGD often degrades pre-trained features, and the privacy composition between pre-training and fine-tuning remains unclear. Developing DP pre-training and privacy-preserving fine-tuning strategies is therefore crucial for enabling private GFMs.

4.4. DP for Agentic Graph Learning

The rapid emergence of LLM-based agents that reason over graph-structured data, spanning knowledge graph reasoning [64,65], GraphRAG pipelines [66,67], and multi-agent coordination [68–71], opens a new frontier that existing DPGL frameworks are ill-equipped to address. Unlike static model training, agentic graph learning involves sequential, multi-step interactions where privacy leakage can compound across tool invocations, graph traversals, and memory retrievals, rendering standard DP composition bounds intractable to maintain. Moreover, agent memory modules that cache intermediate graph states may inadvertently retain sensitive structural information across sessions [72]. Future research should pursue end-to-end DP guarantees over full agent trajectories, private graph memory management strategies, and extended privacy accounting for multi-agent graph coordination protocols.

4.5. Trustworthiness beyond Privacy

Privacy represents only one dimension of trustworthy graph learning in practical deployment scenarios. DP mechanisms may interact in complex ways with robustness [45] and fairness [26], potentially introducing new risks [52], while transparency and interpretability remain largely unexplored in existing DPGL research. Beyond ensuring privacy, jointly achieving robustness, fairness, and other trustworthiness objectives [73] constitutes a key challenge for advancing DPGL toward real-world applications and warrants further systematic investigation.

5. Conclusions

In this survey, we present the first comprehensive review of differentially private graph learning. We propose a systematic taxonomy covering centralized and local DP settings, including node-, edge-, graph-level DP, and LDP-based graph learning methods. Each category summarizes representative techniques, design principles, and practical considerations. Our analysis reveals that effective DPGL methods increasingly move beyond naive noise injection by incorporating privacy-aware strategies, such as sensitivity control, noise calibration, structural information preservation, and adaptive model optimization. Recent advances further exploit graph topology, decoupled learning, and generative modeling to alleviate the utility degradation caused by DP noise while maintaining privacy guarantees. We also discuss open challenges and future research directions, aiming to facilitate the development of scalable, trustworthy, and deployable privacy-preserving graph learning systems.

Author Contributions

L.H.: conceptualization, investigation, methodology, visualization, writing—original draft preparation. L.S.: writing—review and editing. M.L.: writing—review and editing. Y.C.: writing—review and editing. S.S.: supervision, funding acquisition, writing—review and editing. M.Y.: writing—review and editing. J.W.: writing—review and editing. P.L.: supervision. P.S.Y.: supervision. All authors have read and agreed to the published version of the manuscript.

Funding

This work was supported by the National Key Research and Development Program of China (No. 2024YFF0907401), the National Natural Science Foundation of China (No. 62072052), and the BUPT Excellent Ph.D. Students Foundation (No. CX20260030).

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

No new data were created or analyzed in this study. This survey is based on previously published studies cited in the manuscript. Relevant resources and collected references are publicly available through our GitHub repository: https://github.com/thisxyz/Awesome-DPGL (accessed on 20 January 2026).

Conflicts of Interest

The authors declare no conflict of interest.

Use of AI and AI-Assisted Technologies

No AI tools were utilized for this paper.

References

1. Zhu, X.; Tan, V.Y.; Xiao, X. Blink: Link local differential privacy in graph neural networks via Bayesian estimation. In Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security (CCS), Copenhagen, Denmark, 26–30 November 2023; pp. 2651–2664. https://doi.org/10.1145/3576915.3623165.
2. Su, X.; Xue, S.; Liu, F.; et al. A comprehensive survey on community detection with deep learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 4682–4702. https://doi.org/10.1109/TNNLS.2021.3137396.
3. Hu, W.; Fang, H. Towards differential privacy in sequential recommendation: A noisy graph neural network approach. *ACM Trans. Knowl. Discov. Data* **2024**, *18*, 1–21. https://doi.org/10.1145/3643821.
4. Ma, X.; Wu, J.; Xue, S.; et al. A Comprehensive Survey on Graph Anomaly Detection with Deep Learning. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 12012–12038. https://doi.org/10.1109/TKDE.2021.3118815.
5. Mueller, T.T.; Paetzold, J.C.; Prabhakar, C.; et al. Differentially private graph neural networks for whole-graph classification. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 7308–7318. https://doi.org/10.1109/TPAMI.2022.3228315.
6. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. In Proceedings of the International Conference on Learning Representations (ICLR), Toulon, France, 24–26 April 2017; pp. 1–14. https://openreview.net/forum?id=SJU4ayYgl.
7. Wu, F.; Long, Y.; Zhang, C.; et al. Linkteller: Recovering private edges from graph neural networks via influence analysis. In Proceedings of the 2022 IEEE Symposium on Security and Privacy (S&P), San Francisco, CA, USA, 22–26 May 2022; pp. 2005–2024. https://doi.org/10.1109/SP46214.2022.9833806.
8. Wang, X.; Wang, W.H. Group property inference attacks against graph neural networks. In Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security (CCS), Los Angeles, CA, USA, 7–11 November 2022; pp. 2871–2884. https://doi.org/10.1145/3548606.3560662.
9. Meng, L.; Bai, Y.; Chen, Y.; et al. Devil in disguise: Breaching graph neural networks privacy through infiltration. In Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security (CCS), Copenhagen, Denmark, 26–30 November 2023; pp. 1153–1167. https://doi.org/10.1145/3576915.3623173.

10. Dwork, C. Differential privacy: A survey of results. In Proceedings of the International Conference on Theory and Applications of Models of Computation (TAMC), Xi'an, China, 25–29 April 2008; pp. 1–19. https://doi.org/10.1007/978-3-540-79228-4_1.
11. Abadi, M.; Chu, A.; Goodfellow, I.; et al. Deep learning with differential privacy. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS), Vienna, Austria, 24–28 October 2016; pp. 308–318. https://doi.org/10.1145/2976749.2978318.
12. Xiong, X.; Liu, S.; Li, D.; et al. A comprehensive survey on local differential privacy. *Secur. Commun. Netw.* **2020**, *2020*, 8829523. https://doi.org/10.3390/s20247030.
13. Kasiviswanathan, S.P.; Lee, H.K.; Nissim, K.; et al. What can we learn privately? *Siam J. Comput.* **2011**, *40*, 793–826. https://doi.org/10.1137/090756090.
14. Xiang, Z.; Wang, T.; Wang, D. Preserving node-level privacy in graph neural networks. In Proceedings of the 2024 IEEE Symposium on Security and Privacy (S&P), San Francisco, CA, USA, 20–23 May 2024; pp. 4714–4732. https://doi.org/10.1109/SP54263.2024.00270.
15. Kolluri, A.; Baluta, T.; Hooi, B.; et al. LPGNet: Link private graph networks for node classification. In Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security (CCS), Los Angeles, CA, USA, 7–11 November 2022; pp. 1813–1827. https://doi.org/10.1145/3548606.3560705.
16. Li, Y.; Purcell, M.; Rakotoarivelo, T.; et al. Private graph data release: A survey. *Acm Comput. Surv.* **2023**, *55*, 1–39. https://doi.org/10.1145/3569085.
17. Fu, D.; Bao, W.; Maciejewski, R.; et al. Privacy-preserving graph machine learning from data to computation: A survey. *ACM SIGKDD Explor. Newsl.* **2023**, *25*, 54–72. https://doi.org/10.1145/3606274.3606280.
18. Zhang, H.; Wu, B.; Yuan, X.; et al. Trustworthy graph neural networks: Aspects, methods, and trends. *Proc. IEEE* **2024**, *112*, 97–139. https://doi.org/10.1109/JPROC.2024.3369017.
19. Sajadmanesh, S.; Gatica-Perez, D. Locally private graph neural networks. In Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security (CCS), Virtual, 15–19 November 2021; pp. 2130–2145. https://doi.org/10.1145/3460120.3484565.
20. Daigavane, A.; Madan, G.; Sinha, A.; et al. Node-level differentially private graph neural networks. In Proceedings of the ICLR 2022 Workshop on PAIR2Struct: Privacy, Accountability, Interpretability, Robustness, Reasoning on Structured Data (PAIR2Struct), Virtual, 2 April 2022; pp. 1–11. https://openreview.net/forum?id=BCfgOLx3gb9.
21. Lin, W.; Li, B.; Wang, C. Towards private learning on decentralized graphs with local differential privacy. *IEEE Trans. Inf. Forensics Secur.* **2022**, *17*, 2936–2946. https://doi.org/10.1109/TIFS.2022.3198283.
22. Jin, H.; Chen, X. Gromov-wasserstein discrepancy with local differential privacy for distributed structural graphs. In Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence (IJCAI), Vienna, Austria, 23–29 July 2022; pp. 2115–2121. https://doi.org/10.24963/ijcai.2022/294.
23. Olatunji, I.E.; Funke, T.; Khosla, M. Releasing graph neural networks with differential privacy guarantees. *Trans. Mach. Learn. Res.* **2023**, *2023*, 1–29. https://openreview.net/forum?id=wk8oXR0kFA.
24. Sajadmanesh, S.; Shamsabadi, A.S.; Bellet, A.; et al. GAP: Differentially private graph neural networks with aggregation perturbation. In Proceedings of the 32nd USENIX Security Symposium (USENIX Security), 2023, Anaheim, CA, USA, 9–11 August 2023; pp. 3223–3240. https://www.usenix.org/conference/usenixsecurity23/presentation/sajadmanesh.
25. Pei, X.; Deng, X.; Tian, S.; et al. Privacy-enhanced graph neural network for decentralized local graphs. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 1614–1629. https://doi.org/10.1109/TIFS.2023.3329971.
26. Wang, X.; Gu, T.; Bao, X.; et al. Individual fairness for local private graph neural network. *Knowl. Based Syst.* **2023**, *268*, 110490. https://doi.org/10.1016/j.knosys.2023.110490.
27. Chien, E.; Chen, W.N.; Pan, C.; et al. Differentially private decoupled graph convolutions for multigranular topology protection. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 45381–45401. https://openreview.net/forum?id=dd3KNayGFz.
28. Sajadmanesh, S.; Gatica-Perez, D. Progap: Progressive graph neural networks with differential privacy guarantees. In Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM), Merida, Mexico, 4–8 March 2024; pp. 596–605. https://doi.org/10.1145/3616855.3635761.
29. Tang, T.; Niu, Y.; Avestimehr, S.; et al. Edge private graph neural networks with singular value perturbation. *Proc. Priv. Enhancing Technol.* **2024**, *3*, 391–406. https://doi.org/10.56553/popets-2024-0084.
30. Zhang, Q.; Lee, H.k.; Ma, J.; et al. DPAR: Decoupled graph neural networks with node-level differential privacy. In Proceedings of the ACM Web Conference 2024 (WWW), Singapore, 13–17 May 2024; pp. 1170–1181. https://doi.org/10.1145/3589334.3645531.
31. Qi, Y.; Lin, X.; Liu, Z.; et al. LinkGuard: Link locally privacy-preserving graph neural networks with integrated denoising and private learning. In Proceedings of the Companion Proceedings of the ACM Web Conference 2024 (WWW), Singapore, 13–17 May 2024; pp. 593–596. https://doi.org/10.1145/3589335.3651533
32. Ran, X.; Ye, Q.; Hu, H.; et al. Differentially private graph neural networks for link prediction. In Proceedings of the 2024 IEEE 40th International Conference on Data Engineering (ICDE), Utrecht, The Netherlands, 13–16 May 2024; pp. 1632–1644. https://doi.org/10.1109/ICDE60146.2024.00133

33. Zhang, G.; Cheng, X.; Pan, J.; et al. Locally differentially private graph learning on decentralized social graph. *Knowl. Based Syst.* **2024**, *304*, 112488. https://doi.org/10.1016/j.knosys.2024.112488.
34. Hidano, S.; Murakami, T. Degree-preserving randomized response for graph neural networks under local differential privacy. *Trans. Data Priv.* **2024**, *17*, 89–121. http://www.tdp.cat/issues21/abs.a521a23.php.
35. Bhaila, K.; Huang, W.; Wu, Y.; et al. Local differential privacy in graph neural networks: a reconstruction approach. In Proceedings of the 2024 SIAM International Conference on Data Mining (SDM), Houston, TX, USA, 18–20 April 2024; pp. 1–9. https://doi.org/10.1137/1.9781611978032.1.
36. Li, Z.; Li, R.H.; Liao, M.; et al. Privacy-preserving graph embedding based on local differential privacy. In Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM), Boise, ID, USA, 21–25 October 2024; pp. 1316–1325. https://doi.org/10.1145/3627673.3679759.
37. Joshi, R.B.; Indri, P.; Mishra, S. GraphPrivatizer: Improved structural differential privacy for graph neural networks. *Trans. Mach. Learn. Res.* **2024**, *2024*, 1–30. https://openreview.net/forum?id=lcPtUhoGYc
38. Lei, D.; Song, Z.; Yuan, Y.; et al. Achieving personalized privacy-preserving graph neural network via topology awareness. In Proceedings of the ACM on Web Conference 2025 (WWW), Sydney, NSW, Australia, 28 April–2 May 2025; pp. 3552–3560. https://doi.org/10.1145/3696410.3714555.
39. Hou, R.; Ye, Q.; Ran, X.; et al. PrivIM: Differentially private graph neural networks for influence maximization. In Proceedings of the 2025 IEEE 41st International Conference on Data Engineering (ICDE), Hong Kong, China, 19–23 May 2025; pp. 3467–3479. https://doi.org/10.1109/ICDE65448.2025.00259.
40. Zhang, S.; Ye, Q.; Hu, H. Structure-preference enabled graph embedding generation under differential privacy. In Proceedings of the 2025 IEEE 41st International Conference on Data Engineering (ICDE), Hong Kong, China, 19–23 May 2025; pp. 1334–1347. https://doi.org/10.1109/ICDE65448.2025.00104.
41. Zhang, S.; Ye, Q.; Hu, H.; et al. AdvSGM: Differentially private graph learning via adversarial skip-gram model. In Proceedings of the 2025 IEEE 41st International Conference on Data Engineering (ICDE), Hong Kong, China, 19–23 May 2025; pp. 3494–3507. https://doi.org/10.1109/ICDE65448.2025.00261.
42. Wei, J.; Zhu, Y.; Xiao, X.; et al. GCON: Differentially private graph convolutional network via objective perturbation. In Proceedings of the 2025 IEEE 41st International Conference on Data Engineering (ICDE), Hong Kong, China, 19–23 May 2025, pp. 2507–2520. https://doi.org/10.1109/ICDE65448.2025.00189.
43. Li, Y.; Song, X.; Gong, K.; et al. Differentially private graph neural networks for graph classification and its adaptive optimization. *Expert Syst. Appl.* **2025**, *263*, 125798. https://doi.org/10.1016/j.eswa.2024.125798.
44. Li, Y.; Song, X.; Liu, S.; et al. A semi-supervised privacy-preserving graph classification framework enhanced by graph contrastive learning. *Expert Syst. Appl.* **2026**, *299*, 130149. https://doi.org/10.1016/j.eswa.2025.130149.
45. Guo, Z.; Liu, Y.; Ao, X.; et al. GRASP: Differentially private graph reconstruction defense with structured perturbation. In Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2 (SIGKDD), Toronto, ON, Canada, 3–7 August 2025; pp. 767–777. https://doi.org/10.1145/3711896.3736992.
46. Fu, J.; Hong, Y.; Chen, Z.; et al. Safeguarding graph neural networks against topology inference attacks. In Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security (CCS), Taipei, Taiwan, 13–17 October 2025; pp. 2144–2158. https://doi.org/10.1145/3719027.3765173.
47. Yang, C.; Zhu, T.; Liu, Y.; et al. Differentially private adaptive noise for graph neural network in online social networks. *Comput. Netw.* **2025**, *273*, 111757. https://doi.org/10.1016/j.comnet.2025.111757.
48. Xie, Y.; Ding, J.; Xue, P.; et al. Leveraging homophily under local differential privacy for effective graph neural networks. In Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD), Porto, Portugal, 15–19 September 2025; pp. 380–396. https://doi.org/10.1007/978-3-032-06096-9_22.
49. He, L.; Li, C.; Tang, P.; et al. Going deeper into locally differentially private graph neural networks. Forty-second International Conference on Machine Learning (ICML), Vancouver, BC, Canada, 13–19 July 2025; pp. 22548–22565. https://proceedings.mlr.press/v267/he25k.html.
50. Bai, Y.; Xiao, L.; Zhao, H.; et al. DPRO-GNN: Bridging differential privacy and advanced optimization for privacy-preserving graph learning. *Inf. Sci.* **2026**, *723*, 122695. https://doi.org/10.1016/j.ins.2025.122695.
51. Ke, H.; Zhang, S.; Ye, Q.; et al. Adversarial signed graph learning with differential privacy. In Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1 (SIGKDD), Jeju, Republic of Korea, 9–13 August 2026; pp. 532–543. https://doi.org/10.1145/3770854.3780282.
52. He, L.; Li, C.; Tang, P.; et al. Devil's Hand: Data poisoning attacks to locally private graph learning protocols. In Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1 (SIGKDD), Jeju, Republic of Korea, 9–13 August 2026; pp. 395–406. https://doi.org/10.1145/3770854.3780158.
53. He, L.; Tang, P.; Sun, L.; et al. The Devil Within, The Cure Without: Securing Locally Private Graph Learning under Poisoning. In Proceedings of the ACM Web Conference 2026 (WWW), Dubai, United Arab Emirates, 29 June–3 July 2026; pp. 4493–4504. https://doi.org/10.1145/3774904.3792102.
54. Zheng, Y.; Li, C.; Li, Z.; et al. Convergent privacy framework for multi-layer GNNs through contractive message passing. In Proceedings of the Network and Distributed System Security Symposium (NDSS), San Diego, CA, USA, 23–27 February 2026. https://doi.org/10.14722/ndss.2026.240255.

55. He, L.; Tang, P.; Zhang, Y.; et al. Mitigating privacy risks in Retrieval-Augmented Generation via locally private entity perturbation. *Inf. Process. Manag.* **2025**, *62*, 104150. https://doi.org/10.1016/j.ipm.2025.104150.
56. Wu, Z.; Pan, S.; Chen, F.; et al. A comprehensive survey on graph neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 4–24. https://doi.org/10.1109/TNNLS.2020.2978386.
57. He, L.; Wei, Z. Towards structure-aware data augmentation for high-degree graph neural networks. *Inf. Process. Manag.* **2026**, *63*, 104343. https://doi.org/10.1016/j.ipm.2025.104343.
58. Papernot, N.; Abadi, M.; Erlingsson, Ú.; et al. Semi-supervised knowledge transfer for deep learning from private training data. In Proceedings of the International Conference on Learning Representations (ICLR), Toulon, France, 24–26 April 2017. https://openreview.net/forum?id=HkwoSDPgg.
59. Ding, B.; Kulkarni, J.; Yekhanin, S. Collecting telemetry data privately. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 3571–3580. https://doi.org/10.48550/arXiv.1712.01524.
60. Kairouz, P.; Bonawitz, K.; Ramage, D. Discrete distribution estimation under local privacy. In Proceedings of the International Conference on Machine Learning (ICML), New York, NY, USA, 19–24 June 2016; pp. 2436–2444. https://proceedings.mlr.press/v48/kairouz16.html.
61. Wang, N.; Xiao, X.; Yang, Y.; et al. Collecting and analyzing multidimensional data with local differential privacy. In Proceedings of the 2019 IEEE 35th International Conference on Data Engineering (ICDE), Macao, China, 8–11 April 2019; pp. 638–649. https://doi.org/10.1109/ICDE.2019.00063.
62. Li, Z.; Wang, T.; Lopuhaä-Zwakenberg, M.; et al. Estimating numerical distributions under local differential privacy. In Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data (SIGMOD), Virtual, 14–19 June 2020; pp. 621–635. https://doi.org/10.1145/3318464.3389700.
63. Liu, J.; Yang, C.; Lu, Z.; et al. Graph foundation models: Concepts, opportunities and challenges. *IEEE Trans. Pattern Anal. Mach. Intell.* **2025**, *47*, 5023–5044. https://doi.org/10.1109/TPAMI.2025.3548729.
64. Sun, J.; Xu, C.; Tang, L.; et al. Think-on-graph: Deep and responsible reasoning of large language model on knowledge graph. International Conference on Learning Representations (ICLR), Vienna, Austria, 7–11 May 2024; Volume 2024, pp. 3868–3898. https://openreview.net/forum?id=nnVO1PvbTv.
65. Xu, Y.; He, S.; Chen, J.; et al. Generate-on-graph: Treat LLM as both agent and KG for incomplete knowledge graph question answering. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP), Miami, FL, USA, 12–16 November 2024; pp. 18410–18430. https://doi.org/10.18653/v1/2024.emnlp-main.1023.
66. Edge, D.; Trinh, H.; Cheng, N.; et al. From local to global: A graph rag approach to query-focused summarization. *arXiv* **2024**, arXiv:2404.16130. https://doi.org/10.48550/arXiv.2404.16130.
67. Guo, Z.; Xia, L.; Yu, Y.; et al. LightRAG: Simple and fast retrieval-augmented generation. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP (EMNLP Findings), Suzhou, China, 4–9 November 2025; pp. 10746–10761. https://doi.org/10.18653/v1/2025.findings-emnlp.568.
68. Chan, C.M.; Chen, W.; Su, Y.; et al. Chateval: Towards better llm-based evaluators through multi-agent debate. In Proceedings of the International Conference on Learning Representations (ICLR), Vienna, Austria, 7–11 May 2024. https://openreview.net/forum?id=FQepisCUWu.
69. Yang, Y.; Chai, H.; Shao, S.; et al. Agentnet: Decentralized evolutionary coordination for llm-based multi-agent systems. *Adv. Neural Inf. Process. Syst.* **2025**, *38*, 107309–107336. https://openreview.net/forum?id=tXqLxHlb8Z.
70. Zhang, G.; Yue, Y.; Li, Z.; et al. Cut the crap: An economical communication pipeline for llm-based multi-agent systems. In Proceedings of the International Conference on Learning Representations (ICLR), Singapore, 24–28 April 2025. https://openreview.net/forum?id=LkzuPorQ5L.
71. He, L.; He, L.; He, D.; et al. Conflict-resilient multi-agent reasoning via signed graph modeling. *arXiv* **2026**, arXiv:2605.19418. https://doi.org/10.48550/arXiv.2605.19418.
72. Wang, L.; Ma, C.; Feng, X.; et al. A survey on large language model based autonomous agents. *Front. Comput. Sci.* **2024**, *18*, 186345. https://doi.org/10.1007/s11704-024-40231-1.
73. Li, B.; Qi, P.; Liu, B.; et al. Trustworthy AI: From principles to practices. *ACM Comput. Surv.* **2023**, *55*, 177. https://doi.org/10.1145/3555803.