## Supplementary Material

### A. Notations

| Notation | Description |
|---|---|
| $\mathcal{G}$ | Graph dataset |
| $\mathcal{V}$ | Set of nodes |
| $\mathcal{E}$ | Set of edges |
| $\mathbf{X}$ | Matrix of node features |
| $X_u$ | Feature vector of the node $u$ |
| $\hat{\mathcal{E}}$ | Set of edges sampled from $\mathcal{E}$ |
| $\mathcal{N}_u^{out}$ | Out neighbor of the node $u$, $\mathcal{N}_u^{out} = \{v \mid (u,v) \in \mathcal{E}\}$ |
| $\mathcal{N}_u$ | Neighbor of the node $u$, $\mathcal{N}_u = \{v \mid (u,v) \in \mathcal{E}\} \cap \{v \mid (v,u) \in \mathcal{E}\}$ |
| $d_u^{\text{out}}$ | Out degree of the node $u$, $d_u^{\text{out}} = \lvert\mathcal{N}_u^{out}\rvert$ |
| $d_u$ | Degree of the node $u$, $d_u = \lvert\mathcal{N}_u\rvert$ |
| $\hat{\mathcal{G}}$ | Graph dataset with the sampled edge set |
| $(\mathcal{V}, \mathbf{X})$ | Feature part of $\mathcal{G}$ |
| $(\mathcal{V}, \mathcal{E})$ | Graph part of $\mathcal{G}$ |
| $K(\cdot,\cdot)$ | Gaussian kernel function |
| $s_{u,v}$ | Similarity between the node $u$ and the node $v$ calculated by $K(\cdot,\cdot)$ |
| $S$ | Similarity matrix |
| $G_{\mathbf{X}}$ | Weighted graph induced by $(\mathcal{V}, \mathbf{X})$ |
| $\mathcal{F}_u$ | Receptive field of the node $u$ |
| $p(v \mid u)$ | Probability of transitioning from the node $u$ to the node $v$ induced by $(\mathcal{V}, \mathbf{X})$ |
| $M$ | One-step probability transition matrix of the Markov random field induced by $(\mathcal{V}, \mathbf{X})$ |
| $D_t^2(u,u')$ | Diffusion distance between the node $u$ and the node $u'$ induced by $(\mathcal{V}, \mathbf{X})$ |
| $\Psi_t^m$ | Diffusion map induced by $(\mathcal{V}, \mathbf{X})$ |
| $\mathbf{p}(v \mid u)$ | Probability of transitioning from the node $u$ to the node $v$ induced by $(\mathcal{V}, \mathcal{E})$ |
| $\mathbf{M}$ | One-step probability transition matrix of the Markov random field induced by $(\mathcal{V}, \mathcal{E})$ |
| $\mathbf{D}_t^2(u,u')$ | Diffusion distance between the node $u$ and the node $u'$ induced by $(\mathcal{V}, \mathcal{E})$ |
| $\Psi_t'^m$ | Diffusion map induced by $(\mathcal{V}, \mathcal{E})$ |
| $KL(\cdot \mid\mid \cdot)$ | Kullback–Leibler divergence |
| $\mathbf{I}_{u,v}$ | Importance indicator of the edge $(u,v)$ |
| $\rho$ | Sample ratio, $\rho = \frac{\lvert\hat{\mathcal{E}}\rvert}{\lvert\mathcal{E}\rvert}$ |

### B. Additional Datasets details

Here we present the detailed information of the datasets used in our experiments.

**Citeseer** (Giles et al., 1998), **Cora** (McCallum et al., 2000) and **Pubmed** (Sen et al., 2008) are three citation datasets, in which every node represents a publication and every edge represents a citation link. The feature of nodes are one-hot vectors indicating the absence or presence of the corresponding words from the given dictionary. We train models to classify the research topics of publications.

**CoraFull** (Bojchevski & Günnemann, 2017) is another citation dataset, which includes more nodes and edges and its dimension of features is larger than the Cora dataset.

**Amazon** (McAuley et al., 2015) is the Amazon Co-Purchase dataset, where nodes represent merchandises, two merchandises are connected by an edge if they are frequently bought together, node features are the texts merchandise reviews encoded by bag-of-words model, and labels of nodes are the category of merchandises.

**Coauthor**[1] is the Computer Science part of the co-authorship graph based on the Microsoft Academic Graph from the KDD Cup 2016 challenge, where nodes are authors, an edge indicates that two authors have co-authored a paper, features of nodes represent paper keywords for each author's papers, and labels of nodes are the most active fields of study for each author.

[1] https://www.microsoft.com/en-us/research/project/microsoft-academic-graph/

**Gamble** is a financial transaction dataset collected from Alipay platform. In Gamble, every node represents a trade between a payer and a payee. Given two trades, if the payee of the first trade is the payer of the second trade, there is a edge from the first trade to the second trade. We train models to discriminate whether a trade is illegal.

## C. Additional experiments

### C.1 Transductive tasks under low sample ratio

Table 1 summarizes the results of experiments in which the sampling ratio is $20\%$. Where GDS-soft represents the setting of parameters is $(p_1, p_2, q) = (0.1, 0.3, 0.5)$, and GDS-hard represents the setting of parameters is $(p_1, p_2, q) = (0.0, 1.0, 0.8)$. Both of the settings make the expectation of the sampling ratio equal to $20\%$. As discussed in sec.7.2.3, when the sample ratio is low, the soft GDS should perform better than the hard GDS. It is observed that this is true for the Cora and Citeseer datasets, however, it fails in the Pubmed dataset. The reason may be that the average degree $\bar{d} = \frac{2|\mathcal{E}|}{|\mathcal{N}|}$ of nodes in Pubmed is higher than that in Cora and Citeseer as shown in Table 2. Therefore, it is less possible for the hard GDS to generate isolated nodes in the Pubmed dataset than that in the Citeseer and Cora dataset.

| Dataset | Model | US (20%) | GDS-hard | GDS-soft |
|---|---|---|---|---|
| Cora | GCN | 74.76 (2.60) | 73.16 (0.73) | **74.93 (2.43)** |
| | GraphSAGE | 73.33 (4.93) | 70.27 (5.14) | **73.91 (4.31)** |
| | GAT | **73.20 (2.51)** | 70.28 (1.41) | 72.31 (2.30) |
| | JKNet | **73.28 (3.63)** | 71.01 (4.02) | 72.02 (4.82) |
| Citeseer | GCN | 63.51 (3.53) | 48.74 (1.75) | **66.20 (3.18)** |
| | GraphSAGE | 62.62 (5.62) | 49.45 (3.79) | **65.15 (5.09)** |
| | GAT | 61.36 (4.09) | 47.50 (2.01) | **63.93 (3.40)** |
| | JKNet | 60.30 (3.88) | 55.75 (5.85) | **61.80 (5.49)** |
| Pubmed | GCN | 71.53 (2.68) | **76.90 (0.24)** | 73.15 (3.20) |
| | GraphSAGE | 70.49 (5.93) | **74.42 (5.62)** | 71.96 (7.27) |
| | GAT | 70.88 (3.02) | **76.30 (0.98)** | 72.40 (2.75) |
| | JKNet | 72.07 (3.16) | **75.62 (1.39)** | 72.78 (3.06) |

Table 1: **Accuracy of models trained on the subgraphs sampled by different methods under the sample ratio of** $20\%$**.**

| Dataset | $\bar{d}$ | Is the soft GDS better than the hard GDS? |
|---|---|---|
| Citeseer | 2.8 | Yes |
| Cora | 4.0 | Yes |
| Pubmed | 4.4 | No |

Table 2: **The average degree and the comparison of the preformances between the soft and hard GDS.**

### C.2 Detailed results of experiments on inductive tasks

We show detailed results of the inductive tasks. The accuracy of models trained on the full graph is shown in Table 3, the accuracy of models trained on the subgraph sampled by the uniform sampling is shown in Table 4, and the accuracy of models trained on the subgraph sampled by GDS are shown in Table 5. We implement 100 independent tests for each combination of datasets, models, and sampling methods to calculate the mean and standard deviation of the accuracy. For GDS and uniform sampling under different datasets, models and sampling ratios, we bold the results which are better than their counterparts in the other sampling method.

### C.3 Ablation study

In this section, we empirically show that the operations of symmetrization and preventing degeneration indeed work. We compare four versions of GDS: the original GDS, the GDS without sym-

| Dataset | Model | Full Training |
|---|---|---|
| CoraFull | GraphSAGE | 54.28 (2.1) |
| | GAT | 59.79 (0.4) |
| | JKNet | 55.34 (0.4) |
| Amazon | GraphSAGE | 49.50 (14.2) |
| | GAT | 79.76 (2.2) |
| | JKNet | 58.66 (9.9) |
| Coauther | GraphSAGE | 88.00 (1.6) |
| | GAT | 91.23 (0.4) |
| | JKNet | 88.40 (0.3) |
| Gamble | GAT | 95.87 (0.9) |

Table 3: **Accuracy of models trained on the full graph for the inductive tasks.**

| Dataset | Model | US (50%) | US (20%) | US (10%) | US (5%) | US (2%) |
|---|---|---|---|---|---|---|
| CoraFull | GraphSAGE | 54.82 (2.2) | 55.00 (1.8) | 54.51 (1.6) | 52.84 (1.5) | 47.84 (1.1) |
| | GAT | 57.95 (0.5) | 58.93 (4.3) | 55.79 (0.6) | 55.31 (0.6) | 47.96 (0.9) |
| | JKNet | 57.45 (0.5) | 55.92 (0.7) | 53.33 (0.7) | 50.85 (0.8) | 45.75 (1.0) |
| Amazon | GraphSAGE | 46.48 (16.5) | 53.24 (15.8) | **54.61 (17.3)** | 55.27 (17.7) | 52.04 (16.3) |
| | GAT | 82.81 (1.0) | 82.89 (0.6) | **84.33 (0.9)** | **83.78 (0.7)** | **81.41 (0.8)** |
| | JKNet | 64.79 (8.6) | 75.38 (4.3) | 80.69 (2.9) | **84.73 (0.9)** | **82.71 (0.8)** |
| Coauther | GraphSAGE | 88.03 (3.3) | **89.68 (2.3)** | 88.66 (3.0) | 87.91 (1.5) | 88.18 (0.6) |
| | GAT | 90.68 (0.3) | 91.07 (0.3) | 90.54 (0.4) | **89.51 (0.7)** | 86.77 (0.9) |
| | JKNet | 89.18 (0.3) | 89.18 (0.3) | **88.81 (0.3)** | **88.86 (0.3)** | **87.67 (0.5)** |
| Gamble | GAT | 96.14 (0.1) | 94.87 (0.2) | 94.97 (0.2) | 94.15 (0.3) | 93.16 (0.4) |

Table 4: **Accuracy of models trained on the subgraph sampled by the uniform sampling under different sample ratio for the inductive tasks.**

metrization (denote as GDS-non-sym), the GDS without preventing degeneration (denote as GDS-non-prevent) and the GDS without both symmetrization and preventing degeneration (denote as GDS-vanilla). The comparison on the transductive tasks are shown in Table 6, and the comparison on the inductive tasks are shown in Table 7. In the transductive case, we set $(p_1, p_2, q) = (0, 1, 0.5)$ for all the algorithms, and In the inductive case, we set $(p_1, p_2, q) = (0.3, 0.7, 0.5)$ for all the algorithms. It is observed that the original GDS outperforms other versions of GDS in most cases. Noticed that none of the methods can be the best one in all kinds of situations. This indicates that if we control the degree of the symmetrization and preventing degeneration and develop methods to select parameters, we may enhance the GDS's performance. For instance, we can introduce two parameters $\lambda_1 \in [0, 1]$ and $\lambda_2 \in [0, 1]$ to the calculation of the importance indicator

$$\mathbf{I}_{u,v}(\lambda_1, \lambda_2) = \lambda_1 (\frac{1}{d_u \epsilon} + \frac{1}{d_v \epsilon})(\|X_u - X_v\|^2 + \lambda_2 \bar{D}^2) + (1 - \lambda_1) \frac{1}{d_u^{out} \epsilon} (\|X_u - X_v\|^2 + \lambda_2 \bar{D}^2).$$

The values of $\lambda_1$ and $\lambda_2$ may depend on certain properties of the dataset or the GNN model. This will be studied in the further work.

### D. Visualization

In this section, we visualize several graph datasets for a better understanding of the influence of different sampling methods. For visibility, we implement Principal Component Analysis (Wold et al., 1987) to the node features of every dataset and select the first two components as 2-dimension feature for nodes. Then we implement GDS and uniform sampling for the edge set of every dataset. The results are shown in Fig 1, 2, and 3, where the sample ratio of sampling methods are all $10\%$. It observed that compared to the uniform sampling, GDS can preserve more structural information of the original graph. Intuitionally, if the length of an edge is defined as the Euclidean distance between its two end nodes, the longer edges contribute more to sketching the contours of the whole graph. Additionally, if the degrees of two end nodes of an edge are low, this edge is more crucial

| Dataset | Model | GDS (50%) | GDS (20%) | GDS (10%) | GDS (5%) | GDS (2%) |
|---|---|---|---|---|---|---|
| CoraFull | GraphSAGE | **55.17 (2.1)** | **57.09 (1.8)** | **54.92 (1.4)** | **54.43 (1.6)** | **49.39 (1.2)** |
| | GAT | **59.73 (0.4)** | **59.81 (0.5)** | **59.81 (0.5)** | **57.55 (0.6)** | **49.41 (0.9)** |
| | JKNet | **57.82 (0.5)** | **57.82 (0.6)** | **54.02 (0.6)** | **52.52 (0.7)** | **47.34 (0.9)** |
| Amazon | GraphSAGE | **49.40 (17.2)** | **53.67 (16.7)** | 51.93 (18.1) | **56.69 (13.9)** | **56.48 (13.9)** |
| | GAT | **83.04 (1.1)** | **83.69 (1.1)** | 84.25 (0.8) | 84.11 (0.6) | 82.61 (1.0) |
| | JKNet | **69.07 (9.4)** | **79.18 (3.9)** | **83.51 (1.4)** | 84.48 (0.7) | 82.57 (0.7) |
| Coauther | GraphSAGE | **88.35 (2.6)** | 89.67 (1.5) | **88.95 (0.6)** | **88.20 (1.0)** | **88.42 (0.5)** |
| | GAT | **92.15 (0.3)** | **91.53 (0.3)** | **91.62 (0.4)** | 89.06 (0.7) | **88.09 (0.6)** |
| | JKNet | **90.09 (0.3)** | **89.92 (0.3)** | 88.63 (0.3) | 88.35 (0.3) | 87.66 (0.4) |
| Gamble | GAT | **96.17 (0.2)** | **95.11 (0.2)** | **95.04 (0.2)** | **94.49 (0.2)** | **93.69 (0.3)** |

Table 5: **Accuracy of models trained on the subgraph sampled by GDS under different sample ratio for the inductive tasks.**

| Dataset | Model | GDS | GDS-non-sym | GDS-non-prevent | GDS-vanilla |
|---|---|---|---|---|---|
| Cora | GCN | **79.94 (0.5)** | 78.58 (0.4) | 78.52 (0.3) | 77.64 (0.4) |
| | GraphSAGE | **76.63 (5.0)** | 76.06 (5.6) | 76.00 (5.1) | 75.68 (5.5) |
| | GAT | **77.80 (0.9)** | 77.35 (0.8) | 77.04 (1.0) | 76.73 (1.4) |
| | JKNet | 77.88(1.6) | **79.11 (1.8)** | 78.68 (1.4) | 78.24 (2.4) |
| Citeseer | GCN | **70.89 (0.5)** | 69.73 (0.3) | 69.82 (0.4) | 69.95 (0.3) |
| | GraphSAGE | 67.99 (3.8) | 68.88 (4.6) | 67.27 (5.0) | **69.20 (4.8)** |
| | GAT | 68.70 (1.0) | 69.21 (1.2) | 68.83 (1.0) | **69.47 (1.2)** |
| | JKNet | **67.84 (1.7)** | 63.60 (3.9) | 67.53 (1.8) | 64.26 (4.0) |
| Pubmed | GCN | 79.28 (0.4) | 77.20 (0.2) | **79.64 (0.1)** | 77.54 (0.3) |
| | GraphSAGE | **77.43 (6.2)** | 74.62 (5.7) | 76.26 (9.1) | 75.53 (3.4) |
| | GAT | **78.28 (0.7)** | 75.74 (0.7) | 78.20 (0.8) | 75.95 (0.9) |
| | JKNet | **76.57 (1.3)** | 73.88 (2.3) | 76.50 (1.5) | 73.77 (1.9) |

Table 6: **The comparison between different versions of GDS on the transductive tasks**

for constructing the graph, since there are fewer edge that can replace its structural function. These intuitions is corresponding with the importance indicator of GDS.

## References


Aleksandar Bojchevski and Stephan Günnemann. Deep gaussian embedding of graphs: Unsupervised inductive learning via ranking. *arXiv preprint arXiv:1707.03815*, 2017.

C Lee Giles, Kurt D Bollacker, and Steve Lawrence. Citeseer: An automatic citation indexing system. In *Proceedings of the third ACM conference on Digital libraries*, pp. 89–98, 1998.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pp. 43–52, 2015.

Andrew Kachites McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3(2):127–163, 2000.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93–93, 2008.

Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

| Dataset | Model | GDS | GDS-non-sym | GDS-non-prevent | GDS-vanilla |
|---|---|---|---|---|---|
| CoraFull | GraphSAGE | 55.17 (2.1) | **56.71 (2.1)** | 56.42 (2.2) | 55.37 (2.4) |
| | GAT | 59.73 (0.4) | **60.51 (0.5)** | 59.43 (0.5) | 60.00 (0.4) |
| | JKNet | 57.82 (0.5) | **57.99 (0.5)** | 56.37 (0.6) | 57.23 (0.5) |
| Amazon | GraphSAGE | 49.40 (17.2) | 47.84 (16.6) | 47.77 (17.2) | **52.68 (15.9)** |
| | GAT | **83.04 (1.1)** | 81.21 (1.3) | 81.73 (1.2) | 82.12 (1.3) |
| | JKNet | 69.07 (9.4) | 69.32 (7.2) | 69.40 (10.1) | **69.82 (8.0)** |
| Coauther | GraphSAGE | **88.35 (2.6)** | 88.18 (3.6) | 87.28 (3.5) | 87.56 (2.9) |
| | GAT | **92.15 (0.3)** | 91.79 (0.3) | 91.57 (0.3) | 91.18 (0.3) |
| | JKNet | **90.09 (0.3)** | 89.98 (0.3) | 89.07 (0.3) | 89.84 (0.3) |

Table 7: **The comparison between different versions of GDS on the inductive tasks**

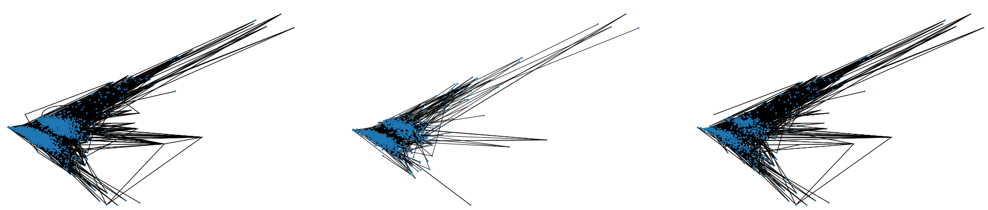

Figure 1: Cora (full graph, sampled by US, sampled by GDS)

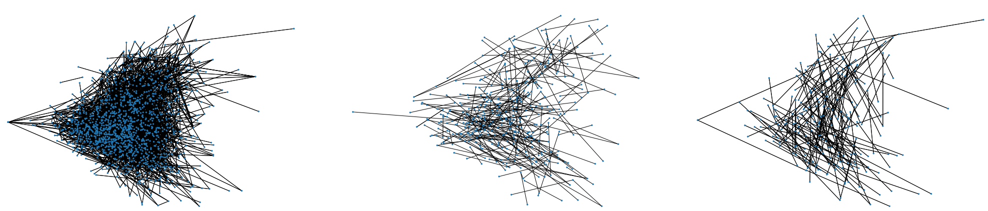

Figure 2: Citeseer (full graph, sampled by US, sampled by GDS)

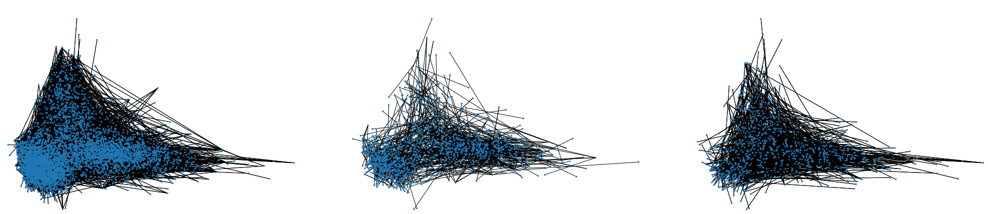

Figure 3: Pubmed (full graph, sampled by US, sampled by GDS)