Improving Graph Contrastive Learning via Adaptive Positive Sampling

Supplementary Material

A. Graph Augmentations and Encoders

Graph Augmentations. Graph augmentation plays a crucial role in enhancing the representation capacity of graph contrastive learning (GCL) by capturing important invariant information. The augmentation techniques used in the baseline GRACE include masking attributes and edge deletion. To be specific, it randomly masks node attributes and deletes edges of initial graphs at each iteration step to generate the augmented graphs.

Encoders. In GCLs, the encoder is responsible for transforming the input graph data into a lower-dimensional feature. The graph encoder in the baseline GRACE consists of a two-layer Graph Convolutional Network (GCN), which is formulated as

$$\mathbf{H} = \sigma(\tilde{\mathbf{A}} \cdot \sigma(\tilde{\mathbf{A}} \cdot \mathbf{X} \cdot \mathbf{W}^{(0)}) \cdot \mathbf{W}^{(1)}). \tag{E.1}$$

where $\sigma(\cdot)$ stands for the nonlinear activation function, such as $\mathrm{ReLU}(\cdot)$. $\tilde{\mathbf{A}}$ denotes the the adjacency matrix subjected to symmetric normalization. $\mathbf{W}^{(i)}$ represents the parameter matrix in the i-th layer.

B. Algorithm Description

The self-expressive learning objective, which is formulated as Equation 5, can be solved via augmented lagrangian multipliers (ALM) [1]. This can be expressed as

$$\begin{aligned}\mathcal{L} =& \|\mathbf{Z} - \mathbf{S}\|_F^2 + \gamma\|\mathbf{S} - \mathbf{CS}\|_F^2 + \lambda\|\mathbf{E}\|_{2,1} \\ &+ \langle \mathbf{H} - \mathbf{ZH} - \mathbf{E}, \mathbf{Y}_1\rangle + \langle \mathbf{S} - \mathbf{C}, \mathbf{Y}_2\rangle \\ &+ \langle \mathbf{C}\mathbf{1}_n - \mathbf{1}_n, \mathbf{Y}_3\rangle + \langle \mathbf{S} - \mathbf{D}, \mathbf{Y}_4\rangle \\ &+ \mu/2 \left(\|\mathbf{H} - \mathbf{ZH} - \mathbf{E}\|_F^2 + \|\mathbf{S} - \mathbf{C}\|_F^2\right. \\ &\left.+ \|\mathbf{C}\mathbf{1}_n - \mathbf{1}_n\|_F^2 + \|\mathbf{S} - \mathbf{D}\|_F^2\right)\end{aligned} \tag{E.2}$$

where $\langle \mathbf{X}, \mathbf{Y}\rangle = \mathrm{Tr}(\mathbf{X}\mathbf{Y}^\top)$ terms the inner product of matrix $\mathbf{X}$ and $\mathbf{Y}$, and $\mathbf{C}$ and $\mathbf{D}$ denote two auxiliary variables, which is introduced to enhance the representation ability of the matrix $\mathbf{S}$. And $\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3$ and $\mathbf{Y}_4$ represents four Lagrangian multiplier variables. $\mu > 0$ denotes an additional parameter.

The above objective function can be optimized via Alternating Direction Methods of Multipliers (ADMM) [2]. ADMM alternately updates $\mathbf{Z}, \mathbf{S}, \mathbf{C}, \mathbf{D}, \mathbf{E}$ alternately while fixing others. The details are presented in Algorithm 1.

Algorithm 1: Self-expressive Learning Framework

Data: Node features $\mathbf{H} \in \mathbb{R}^{n\times d}$;
Input: The number of blocks k, parameters γ and λ, the maximum number of iterations $\#maxI$;

% Initialization %
Initialize $l = 0, \mu^l = 10^{-6}, \mu_{max} = 10^4, \rho = 1.1, \varepsilon = 10^{-7}$, and $\mathbf{Y}_1^0 = \mathbf{Y}_2^0 = \mathbf{Y}_3^0 = \mathbf{0}$, and randomly initialize $\mathbf{Z}^0, \mathbf{S}^0, \mathbf{C}^0, \mathbf{D}^0 \in \mathbb{R}^{n\times n}$.
while $\|\mathbf{H} - \mathbf{Z}^l\mathbf{H} - \mathbf{E}^l\|_\infty > \varepsilon, \|\mathbf{S}^l - \mathbf{C}^l\|_\infty > \varepsilon, \|\mathbf{S}^l - \mathbf{D}^l\|_\infty > \varepsilon, \|\mathbf{C}^l\mathbf{1}_n - \mathbf{1}_n\|_\infty > \varepsilon$ *and* $l < \#maxI$ **do**
  % Update coefficient matirices %
  Update coefficient matirices $\mathbf{Z}^{l+1}$ and $\mathbf{S}^{l+1}$ via
  $\mathbf{Z}^{l+1} = \mathrm{argmin}_{\mathbf{Z}} \left\|\mathbf{Z} - \mathbf{S}^l\right\|_F^2 + \mu^l/2 \left\|\mathbf{H} - \mathbf{ZH} - \mathbf{E}^l + \mathbf{Y}_1^l/\mu^l\right\|_F^2$ and
  $\mathbf{S}^{l+1} = \mathrm{argmin}_{\mathbf{S}} \left\|\mathbf{Z}^{l+1} - \mathbf{S}\right\|_F^2 + \gamma \left\|\mathbf{S} - \mathbf{C}^l\mathbf{S}\right\|_F^2 + \frac{\mu^l}{2}\left(\left\|\mathbf{S} - \mathbf{C}^l + \mathbf{Y}_2^l/\mu^l\right\|_F^2 + \left\|\mathbf{S} - \mathbf{D}^l + \mathbf{Y}_4^l/\mu^h\right\|_F^2\right)$.
  Since the nonnegative and symmetric constraints on $\mathbf{S}$, let $\mathbf{S}^{l+1} = \max(\mathbf{S}^{l+1}, \mathbf{0})$, and $\mathbf{S}^{l+1} = \left(\mathbf{S}^{l+1} + (\mathbf{S}^{l+1})^\top\right)/2$.
  % Update auxiliary variables %
  Update auxiliary variables $\mathbf{C}^{l+1}$ and $\mathbf{D}^{l+1}$ via
  $\mathbf{C}^{l+1} = \mathrm{argmin}_{\mathbf{C}}\, \gamma \left\|\mathbf{S}^{l+1} - \mathbf{CS}^{l+1}\right\|_F^2 + \mu^l/2 \left(\left\|\mathbf{S}^{l+1} - \mathbf{C} + \mathbf{Y}_2^l/\mu^l\right\|_F^2\right.$
  and $\mathbf{D}^{l+1} = \mathrm{argmin}_{\mathbf{D}} \|\mathbf{D} - \mathbf{T}\|_F^2$, s.t. $\mathrm{Tr}(\mathbf{D}) = k$, where $\mathbf{T} = \mathbf{S}^{l+1} + \mathbf{Y}^l/\mu^l$.
  % Update error %
  Update error $\mathbf{E}^{l+1}$ via
  $\underset{\mathbf{E}}{\mathrm{argmin}}\, \lambda\|\mathbf{E}\|_{2,1} + \frac{\mu^l}{2}\left\|\mathbf{H} - \mathbf{Z}^{l+1}\mathbf{H} - \mathbf{E} + \frac{\mathbf{Y}_1^l}{\mu^l}\right\|_F^2$.
  % Update parameters %
  Update $\mathbf{Y}_1^{l+1}, \mathbf{Y}_2^{l+1}, \mathbf{Y}_3^{l+1}, \mathbf{Y}_4^{l+1}$ and μ^{l+1} via
  $\mathbf{Y}_1^{l+1} = \mathbf{Y}_1^l + \mu^l\left(\mathbf{H} - \mathbf{Z}^{l+1}\mathbf{H} - \mathbf{E}^{l+1}\right)$,
  $\mathbf{Y}_2^{l+1} = \mathbf{Y}_2^l + \mu^l\left(\mathbf{S}^{l+1} - \mathbf{C}^{l+1}\right)$,
  $\mathbf{Y}_3^{l+1} = \mathbf{Y}_3^l + \mu^l\left(\mathbf{C}^{l+1}\mathbf{1}_n - \mathbf{1}_n\right)$,
  $\mathbf{Y}_4^{l+1} = \mathbf{Y}_4^l + \mu^l\left(\mathbf{S}^{l+1} - \mathbf{D}^{l+1}\right)$,
  $\mu^{l+1} = \min\left(\mu_{\max}, \rho\mu^l\right)$, respectively.
end
return $\mathbf{Z}^* = \mathbf{Z}^l, \mathbf{S}^* = \mathbf{S}^l$

C. Proof for Theorem 1

Theorem 1. *The affinity matrix* $\mathbf{M}$*, which is generated by optimizing Equation (5), satisfies the BDP in Definition 1.*

Proof. First of all, the objective function falls into the general subspace clustering problem, that is,

$$\min_{\mathbf{Z}} \Phi(\mathbf{Z}, \mathbf{H}) \quad \text{s.t.} \mathbf{H} = \mathbf{ZH} \tag{E.3}$$

Secondly, based on the Enforced Block Diagonal (EBD)

condition in [3], it is not difficult to verify that for any permutation matrix $\mathbf{P}$, the above equation satisfies $\Phi(\mathbf{Z}, \mathbf{H}) = \Phi(\mathbf{PZP}^\top, \mathbf{PH})$.

Finally, each section of the above equation has a unique solution, which is presented in Algorithm 1. It indicates that this equation has a unique solution $\mathbf{M}$. To sum up, based on the Theorem 3 in [3] and the above explanations, it is concluded that the affinity matrix $\mathbf{M}$, which is obtained by optimizing Equation 5, is block-diagonal. □

D. Proof for Theorem 2

Theorem 2. *The contrastive loss of HEATS ($\mathcal{L}_{heats}$) is a more stringent estimate of mutual information (MI) between node attributes and embeddings than that of the local baseline HomoGCL, that is,*

$$\mathcal{L}_{homogcl} \leq \mathcal{L}_{heats} \leq I(\mathbf{X}; \mathbf{H}, \tilde{\mathbf{H}}), \tag{E.4}$$

where $\mathbf{X}$ denotes the node attributes, and $\mathbf{H}$ and $\tilde{\mathbf{H}}$ represent the node embeddings in two augmented views.

Proof. First of all, the contrastive loss of HEATS ($\mathcal{L}_{heats}$) is a lower bound of MI between node attributes and embeddings, i.e., $\mathcal{L}_{heats} \leq I(\mathbf{X}; \mathbf{H}, \tilde{\mathbf{H}})$, as proved below.

The InfoNCE objective can be formulated as

$$I_{\text{NCE}}(\mathbf{H}; \tilde{\mathbf{H}}) \triangleq \mathbb{E}\left[\frac{1}{n}\sum_{v\in V}\log\frac{e^{\theta(\mathbf{h}_v, \tilde{\mathbf{h}}_v)}}{\frac{1}{n}\sum_{u\in V} e^{\theta(\mathbf{h}_v, \tilde{\mathbf{h}}_u)}}\right], \tag{E.5}$$

where the expectation is over n nodes from the joint distribution $\prod_v p(\mathbf{h}_v, \tilde{\mathbf{h}}_v)$. According to the conclusion in HomoGCL, the InfoNCE is a lower bound of MI, i.e.,

$$I_{\text{NCE}}(\mathbf{H}; \tilde{\mathbf{H}}) \leq I(\mathbf{X}; \mathbf{H}, \tilde{\mathbf{H}}) \tag{E.6}$$

For the contrastive loss of HEATS ($\mathcal{L}_{heats}$), as defined in Equation 7, it consists of two parts: $\mathcal{L}^{(1)}_{heats}$ and $\mathcal{L}^{(2)}_{heats}$.

$$\begin{aligned}\mathcal{L}_{heats} &= \frac{1}{2n}\sum_{v\in V}\Big(\ell_{ht}(\mathbf{h}_v, \tilde{\mathbf{h}}_v) + \ell_{ht}(\tilde{\mathbf{h}}_v, \mathbf{h}_v)\Big) \\ &= \frac{1}{2}(\mathcal{L}^{(1)}_{heats} + \mathcal{L}^{(2)}_{heats})\end{aligned} \tag{E.7}$$

Let's start with one part

$$\mathcal{L}^{(1)}_{heats} = \mathbb{E}\left[\frac{1}{n}\sum_{v\in V}\log\frac{\text{po}_v}{\text{po}_v + \text{ne}_v}\right], \tag{E.8}$$

$$\text{po}_v = \text{e}^{\frac{\theta(\mathbf{h}_v, \tilde{\mathbf{h}}_v)}{\tau}} + \sum_{u\in\mathcal{P}^M_v} m_{v,u}\cdot\text{e}^{\frac{\theta(\mathbf{h}_v, \mathbf{h}_u)}{\tau}}, \tag{E.9}$$

$$\text{ne}_v = \sum_{t\in\mathcal{N}^M_v}\text{e}^{\frac{\theta(\mathbf{h}_v, \tilde{\mathbf{h}}_t)}{\tau}} + \sum_{t\in\mathcal{N}^M_v}\text{e}^{\frac{\theta(\mathbf{h}_v, \mathbf{h}_t)}{\tau}}. \tag{E.10}$$

Given the BDP and idempotent property, along with the proposed sparsification operation, the number of positive samples is far less than that of nodes, i.e., $|\mathcal{P}^M_v| \ll n$. Due to any $m_{v,u} \in [0, 1]$, setting the temperature coefficient $\tau = 1$ for convenience, we have

$$\text{po}_v \leq n\cdot\text{e}^{\theta(\mathbf{h}_v, \tilde{\mathbf{h}}_u)}, \tag{E.11}$$

and

$$\text{ne}_v \approx \sum_{u\in V}\text{e}^{\theta(\mathbf{h}_v, \mathbf{h}_u)} + \sum_{u\in V}\text{e}^{\theta(\mathbf{h}_v, \tilde{\mathbf{h}}_u)} \tag{E.12}$$

Moreover, combining Equation E.5, Equation E.11, Equation E.12, and

$$\mathbb{E}\left[\frac{1}{n}\sum_{v\in V}\log\frac{\text{po}_v}{\text{po}_v + \text{ne}_v}\right] \leq \mathbb{E}\left[\frac{1}{n}\sum_{v\in V}\log\frac{\text{po}_v}{\text{ne}_v}\right], \tag{E.13}$$

it can be derived that

$$\begin{aligned}\mathcal{L}^{(1)}_{heats} &\leq \mathbb{E}\left[\frac{1}{n}\sum_{v\in V}\log\frac{n\cdot e^{\theta(\mathbf{h}_v, \tilde{\mathbf{h}}_v)}}{\sum_{u\in V} e^{\theta(\mathbf{h}_v, \tilde{\mathbf{h}}_u)}}\right] \\ &= I_{\text{NCE}}(\mathbf{H}, \tilde{\mathbf{H}}).\end{aligned} \tag{E.14}$$

Similarly, we can get that $\mathcal{L}^{(2)}_{heats} \leq I_{\text{NCE}}(\tilde{\mathbf{H}}, \mathbf{H})$. Accordingly, there is

$$\mathcal{L}_{heats} \leq \frac{1}{2}(I_{\text{NCE}}(\mathbf{H}, \tilde{\mathbf{H}}) + I_{\text{NCE}}(\tilde{\mathbf{H}}, \mathbf{H})) = I_{\text{NCE}}(\mathbf{H}, \tilde{\mathbf{H}}) \tag{E.15}$$

By combining Equation E.15 and Equation E.6, we establish the inequality $\mathcal{L}_{heats} \leq I(\mathbf{X}; \mathbf{H}, \tilde{\mathbf{H}})$.

Secondly, $\mathcal{L}_{heats}$ is a stricter lower bound of $I(\mathbf{X}; \mathbf{H}, \tilde{\mathbf{H}})$ than $\mathcal{L}_{homogcl}$, as described below. Similar to HEATS, HomoGCL is implemented based on GRACE, and can be formulated as

$$\mathcal{L}_{homogcl} = \frac{1}{2n}\sum_{v\in V}(\ell_{ho}(\mathbf{h}_v, \tilde{\mathbf{h}}_v) + \ell_{ho}(\tilde{\mathbf{h}}_v, \mathbf{h}_v)) \tag{E.16}$$

where

$$\ell_{ho}(\mathbf{h}_v, \tilde{\mathbf{h}}_v) = \sum_{v\in V}\log\frac{\text{pos}}{\text{pos} + \text{neg}} \tag{E.17}$$

$$\text{pos} = \text{e}^{\frac{\theta(\mathbf{h}_v, \tilde{\mathbf{h}}_v)}{\tau}} + \sum_{u\in N(v)} s_{v,u}\cdot\text{e}^{\frac{\theta(\mathbf{h}_v, \mathbf{h}_u)}{\tau}}, \tag{E.18}$$

where $N(v)$ represents the neighbor node set of node v and $s_{v,u} \in [0, 1]$ stands for the saliency value.

$$\text{neg} = \sum_{t\in\{V\setminus N(v)\}}\text{e}^{\frac{\theta(\mathbf{h}_v, \tilde{\mathbf{h}}_t)}{\tau}} + \sum_{t\in\{V\setminus N(v)\}}\text{e}^{\frac{\theta(\mathbf{h}_v, \mathbf{h}_t)}{\tau}}. \tag{E.19}$$

Given that the graph is sparse, we have that $|N(v)| \leq |\mathcal{P}_v^M|$, which implies $\mathrm{pos} \leq \mathrm{po}_v$. Additionally, it holds that $\mathrm{neg} \geq \mathrm{ne}_v$.

Therefore, it can be deduced that

$$\ell_{ho}(\mathbf{h}_v, \tilde{\mathbf{h}}_v) \leq \ell_{ht}(\mathbf{h}_v, \tilde{\mathbf{h}}_v) \tag{E.20}$$

and

$$\ell_{ho}(\tilde{\mathbf{h}}_v, \mathbf{h}_v) \leq \ell_{ht}(\tilde{\mathbf{h}}_v, \mathbf{h}_v). \tag{E.21}$$

Thus, there is

$$\mathcal{L}_{homogcl} \leq \mathcal{L}_{heats} \tag{E.22}$$

In conclusion, we have $\mathcal{L}_{homogcl} \leq \mathcal{L}_{heats} \leq I(\mathbf{X}; \mathbf{H}, \tilde{\mathbf{H}})$. □

E. Datasets Description and Statistics

This section presents the details of the benchmark datasets, which consist of twelve graphs and three image datasets.

E.1. Graphs

Citation network. Cora, CiteSeer, and PubMed are the three benchmark graph datasets on citation networks [4]. The nodes represent academic papers, edges indicate citation relationships between papers, node features are encoded using bag-of-words representations, and node labels indicate the academic topics of papers.

Reference network. Wiki-CS is a reference network from Wikipedia categories [5]. The nodes represent Computer Science articles, edges indicate the hyperlinks between articles, node features are calculated as the average of pre-trained GloVe word embeddings, and the node labels correspond to the branches of computer science.

Co-purchase networks. Amazon Photo (short as Photo) and Amazon Computers (short as Computers) are two co-purchase networks collected from Amazon [6]. Nodes represent products, edges represent the co-purchased relations of products, and node features are bag-of-words vectors extracted from product reviews.

WebKB. Cornell, Texas, and Wisconsin are three webpage datasets[1], which are collected from the computer science departments of their namesake universities. The nodes stand for webpages, and the edges stand for hyperlinks between them. Node features are the bag-of-words representation of webpages. These webpages are manually classified into five categories, student, project, course, staff, and faculty.

Wikipedia network. Chameleon and squirrel are two networks on specific topics in Wikipedia [7]. The nodes represent webpages and the edges are links between webpages. Node features correspond to several informative nouns in the web pages, and labels represent the average monthly traffic of the web pages.

[1] http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-11/www/wwkb/

Co-occurrence network. Actor is an actor co-occurrence network [8]. The nodes corresponding to actors and edges denote co-occurrence on the Wikipedia page. Node features terms the keywords in these pages. Labels are the words on the actor's Wikipedia.

E.2. Images

CIFAR-10, CIFAR-100, and STL-10 are three image benchmark datasets in computer vision. Specifically, CIFAR-10 and CIFAR-100 are designed by the Canadian Institute for Advanced Research (CIFAR), while STL-10 is developed by Stanford University. CIFAR-10 contains 60,000 images of 32x32 pixels, categorized into 10 classes, such as airplane and automobile. CIFAR-100 expands it by introducing a taxonomy of 100 classes, organized hierarchically into 20 superclasses, each containing 5 subclasses. STL-10 consists of 60,000 images of 96x96 pixels, which can be classified into 10 distinct classes.

F. Introduction of Baselines

Graph Neural Networks (GNNs). GCN, a well-recognized semi-supervised GNN, learns node representations by emphasizing the similarity of the adjacent nodes. GAT, a spatial variant of GCN, introduces a self-attention mechanism for flexible capturing of the relationships between nodes. JKNet innovatively integrates representations across layers to form the final representations, rendering it more flexible and effective than conventional stacked GCN.

Unsupervised GNNs. (1) Network Embedding models. DeepWalk and Node2Vec are network embedding models that employ Random Walk strategies to establish the proximity information in graph structures. Furthermore, DeepWalk emphasizes the analogy of random walk paths to sentences to leverage word2vec [9] algorithm, while Node2Vec enhances the flexibility of the random walk process through breadth-first search (BFS) and depth-first search (DFS).

(2) Graph Generative models. GAE and VGAE are both graph-generative models. The former encodes graph structures into a compact latent space via an autoencoder for efficient graph reconstruction, while the latter introduces a variational approach with a Gaussian distribution to capture the uncertainty in the graph structure.

(3) Graph Contrastive Learning models. DGI, MVGRL, GRACE, GCA, BGRL, and HomoGCL are common baseline GCL models. Specifically, DGI brings in a local-global contrastive learning framework by maximizing the mutual information between the node-level and the graph-level representations. MVGRL extends the DGI framework by introducing graph diffusion techniques and a multi-view mechanism. Inspired by the two-branch contrastive learning (CL) framework, which is widely employed in computer vision, GRACE extends it to the graph domain via graph encoders and graph augmentations such as edge deletion and feature

masking. Moreover, GCA innovates by incorporating adaptive augmentation strategies tailored to the graph structure. Besides, BGRL enhances GCL models through the incorporation of the bootstrapped strategy, namely bootstrapping the output of a delayed version of the encoder. SELENE integrates node attributes and network structure information to alleviate the impact of decreasing network homophily ratio. In addition, HomoGCL capitalizes on graph homophily to expand the positive set using the neighbor nodes that possess saliency.

G. Experiment setups of image classification

First, the network structure of the proposed variant models is designed to be consistent with that of the baseline models. The backbones of the baseline model are ResNet-18 and ResNet-50. The experiments are conducted on three public image datasets CIFAR-10, STL-10, and CIFAR-100. Secondly, the network parameters are updated by Adam optimizer with the learning rate is 0.0003 and the weight decay rate is 10^{-6}. The dimensions of the hidden layers and the projection layers are set to 64, and the batch size is set to 128. The temperature parameter τ is chosen from $\{0.1, 0.2, 0.3, 0.4, 0.5\}$. Thirdly, for the parameters γ and λ, which is introduced by the proposed framework, the values are selected from the set $\{10^{-2}, 10^{-3}, 10^{-4}\}$. For the CIFAR-10 and STL-10 datasets, the block parameter k is selected from the set containing all values less than the number of classes. For the CIFAR-100, the selection of k is among $\{10, 20, 30, 40, 50, 60, 70\}$. For the combination of these hyperparameters, the grid search strategy is used.

References

[1] Zhouchen Lin, Minming Chen, and Yi Ma. The Augmented Lagrange Multiplier Method for Exact Recovery of Corrupted Low-Rank Matrices. *CoRR*, abs/1009.5055, 2010. 1

[2] Yu Wang, Wotao Yin, and Jinshan Zeng. Global convergence of admm in nonconvex nonsmooth optimization. *Journal of Scientific Computing*, 78:29–63, 2019. 1

[3] Canyi Lu, Jiashi Feng, Zhouchen Lin, Tao Mei, and Shuicheng Yan. Subspace clustering by block diagonal representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):487–501, 2019. 2

[4] 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. 3

[5] P Mernyei and C Wiki-CS Cangea. A wikipedia-based benchmark for graph neural networks. arxiv 2020. *arXiv preprint arXiv:2007.02901*, 2007. 3

[6] Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation, 2019. 3

[7] Benedek Rozemberczki, Carl Allen, and Rik Sarkar. Multi-scale attributed node embedding. *Journal of Complex Networks*, 9(2):cnab014, 2021. 3

[8] Jie Tang, Jimeng Sun, Chi Wang, and Zi Yang. Social influence analysis in large-scale networks. In *SIGKDD*, pages 807–816. ACM, 2009. 3

[9] Yoav Goldberg and Omer Levy. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *CoRR*, abs/1402.3722, 2014. 3