Appendix

1 EXPERIMENTS

1.1 Dataset Description

ITL A dataset derived from connectivity data during sensor information transition, collected from [8]. Vertices represent distinct sensors, while edges depict the transmission of messages between them. The edge probability $P(e)$ indicates the likelihood of a message being successfully transmitted from a sender sensor to the receiver.

KRC A Protein-Protein Interaction (PPI) network dataset from [5]. Nodes correspond to individual proteins, and edges denote interactions between protein pairs. Here, $P(e)$ represents the confidence level of an interaction between two proteins.

DBLP A citation network dataset from [1], wherein each node symbolizes a paper and each edge indicates a citation relationship between two papers. The edge probability $P(e)$ is determined by $(20, 10^{-3})$-*obfuscation* [2] following [7, 9, 10].

Road Networks The Beijing (BJ) [3] and California (CAL) [6] datasets pertain to road networks. Intersections and endpoints are denoted by nodes, while roads connecting these are represented by edges. In the CAL dataset, $P(e)$ is injected by $(20, 10^{-3})$-*obfuscation* [2] following [7, 9, 10].

1.2 Inductive Test

We also evaluated the inductive learning performance of UnG-MoCha. Firstly, we pre-train our model on KRC dataset with only small size motifs as training data, then fine-tune the pre-trained model with different number of training data under three settings: predict the count of large size motifs on KRC, it of small size motifs on DBLP and large size motifs on DBLP. The result is visualized in Figure 1. With the increase of training data, the pre-trained model achieves better performance. With 20% of training data, the fine-tuned model can achieve acceptable results. It is interesting that the pre-trained model achieves a relatively lower MAPE when estimating the counts of small-size motifs on an unseen dataset, demonstrating UnG-MoCha is less impacted by the input graph. The result indicates that UnG-MoCha can be applied to handle large-scale graphs after pre-trained on smaller-size datasets.

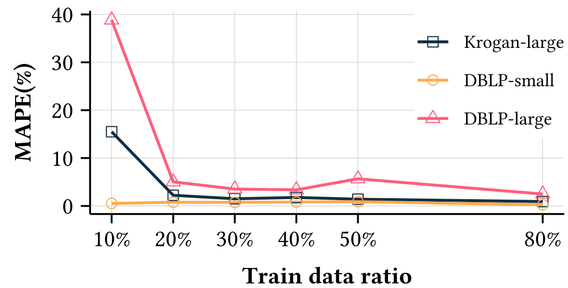

Figure 1: Results of inductive test.

2 PROOF OF THEOREM 1

We first recall the loss function of UnG-MoCha:

$$\mathcal{L}_{\Theta}(\mathcal{M}) = \alpha\mathcal{L}(m,\hat{m}) + (1-\alpha)\mathcal{L}(v,\hat{v}) + \gamma\mathcal{L}_{CCA} \tag{1}$$

where $\mathcal{L}(m,\hat{m})$ is:

$$\mathcal{L}(m,\hat{m}) = \|m-\hat{m}\|^2 \tag{2}$$

where $\hat{m} = MLP([\mathbf{h}_{\mathcal{M}}\|\mathbf{h}_{\mathcal{G}}])$.

The definition of multi-layer perceptron(MLP) is below:

Definition 2.1 (multi-layer perceptron [11]). *A K-layer multi-layer perceptron $f_{MLP}:\mathbb{R}^d \to \mathbb{R}^n$ is the function*

$$f_{MLP}(x) = T_K \circ \rho_K \circ \cdots \circ \rho_1 \circ T_1(x) \tag{3}$$

where $T_k : x \mapsto W_k x + b_k$ is an affine function and $\rho_k : x \mapsto (g_k(x))$ is the non-linear activation function.

Similar to $\mathcal{L}(m,\hat{m})$, $\mathcal{L}(v,\hat{v})$ is:

$$\mathcal{L}(v,\hat{v}) = \|v-\hat{v}\|^2, \tag{4}$$

and $\hat{v}$ is obtained from the same multi-layer perceptron MLP.

$\mathcal{L}_{CCA}$ is written as:

$$\begin{aligned}&\mathcal{L}_{dist}(MLP(\mathbf{h}_{\mathcal{M}}), MLP(\mathbf{h}_{\mathcal{G}}))\\ &+\lambda(\mathcal{L}_{dl}(MLP(\mathbf{h}_{\mathcal{M}})) + \mathcal{L}_{dl}(MLP(\mathbf{h}_{\mathcal{G}})))\end{aligned} \tag{5}$$

Correlation loss $\mathcal{L}_{dist}$ is

$$\mathcal{L}_{dist} = \frac{1}{2}\|MLP(\mathbf{h}_{\mathcal{M}}) - MLP(\mathbf{h}_{\mathcal{G}})\|_F^2, \tag{6}$$

and decorrelation loss $\mathcal{L}_{dl}$ is

$$\mathcal{L}_{dl}(\mathbf{U}) = \|\mathbf{U}^T\mathbf{U} - I\|_F^2 \tag{7}$$

To obtain the upper bound of estimation error, the upper bound of every component of Equation (1) needs to be derived. Firstly, we derive the upper bound of $\mathcal{L}(m,\hat{m})$:

$$\begin{aligned}\mathcal{L}(m,\hat{m}) &= \|m-\hat{m}\|^2\\ &= \|m - MLP([\mathbf{h}_{\mathcal{M}}\|\mathbf{h}_{\mathcal{G}}])\|^2\end{aligned} \tag{8}$$

Assume that all the operations in MLP are all locally Lipschitz-continuous, and that their partial derivatives $\partial g_k(x)$ can be computed and efficiently maximized.

Theorem 2.1 (Rademacher Theorem [4]). *If $f:\mathbb{R}^d \to \mathbb{R}^n$ is a locally Lipschitz continuous function, then f is differentiable almost everywhere. Moreover, if f is Lipschitz continuous, then*

$$\mathcal{L}_f \le \sup_{x\in R^d} \|\nabla f(x))\|_2. \tag{9}$$

With Theorem 2.1 and the assumption, the MLP can be considered as locally Lipschitz-continuous and the upper bound of each component in Equation (1) can be derived. With the definition of

MLP and the assumption, it can be derived that

$$\begin{aligned}\mathcal{L}(m,\hat{m}) \le \frac{\partial \mathcal{L}(m,\hat{m})}{\partial(\mathbf{h}_{\mathcal{M}}||\mathbf{h}_{\mathcal{G}})} &= \frac{\partial \|m - MLP([\mathbf{h}_{\mathcal{M}}||\mathbf{h}_{\mathcal{G}}])\|^2}{\partial(\mathbf{h}_{\mathcal{M}}||\mathbf{h}_{\mathcal{G}})} \\ &\le 2(m - MLP([\mathbf{h}_{\mathcal{M}}||\mathbf{h}_{\mathcal{G}}]))\prod_{k=1}^{K}\|W_k\|_2\end{aligned} \tag{10}$$

where K is the layer number of multi-layer perceptron.

Substitute Equation (10) into Equation (2), Equation (2) can be rewritten as:

$$\|m-\hat{m}\|^2 \le 2(m - MLP([\mathbf{h}_{\mathcal{M}}||\mathbf{h}_{\mathcal{G}}]))\prod_{k=1}^{K}\|W_k\|_2 \tag{11}$$

Through Equation (11), the bound of $\hat{m}$ can be derived as:

$$m - 2\tau \le \hat{m} \le \tau, \tag{12}$$

where $\tau \in \mathbb{R}$ is a constant and $\tau \le \prod_{k=1}^{K}\|W_k\|_2$.

Similarly, we can get the upper bound of $\mathcal{L}(v,\hat{v})$, which is

$$\mathcal{L}(v,\hat{v}) \le 2(v-\epsilon)\prod_{k=1}^{K}\|W_k\|_2 \tag{13}$$

The bound of $\hat{v}$ is:

$$v - 2\tau \le \hat{v} \le \tau \tag{14}$$

For $\mathcal{L}_{CCA}$, we prove the upper bound from variance-covariance perspective. After transformation by MLP, $\mathbf{h}_{\mathcal{M}}$ and $\mathbf{h}_{\mathcal{G}}$ have the same dimension and can be considered as augmented by $s \sim p_{aug}(x)$. $\widetilde{z}$ denotes the representation of s. Correlation loss $\mathcal{L}_{dist}$ can be written as:

$$\begin{aligned}\mathcal{L}_{dist} &= \frac{1}{2}\|MLP(\mathbf{h}_{\mathcal{M}}) - MLP(\mathbf{h}_{\mathcal{G}})\|_F^2 \\ &= \sum_{i=1}^{N}\sum_{j=1}^{D}(\widetilde{z}_{i,j}^{\mathbf{h}_{\mathcal{M}}} - \widetilde{z}_{i,j}^{\mathbf{h}_{\mathcal{G}}})^2 \\ &\cong N * \mathbb{E}_{\mathbf{h}_{\mathcal{M}};\mathbf{h}_{\mathcal{G}}}(\sum_{k=1}^{D}\mathbb{V}_s[\widetilde{z}_k])\end{aligned} \tag{15}$$

The decorrelation loss can be transformed to the sum of Pearson correlation coefficient [12]:

$$\begin{aligned}\mathcal{L}_{dl}(U) &= \|U^TU - I\|_F^2 \\ &= \|Cov[U] - I\|_F^2 \\ &\cong \sum_{i\ne j}\rho_{i,j}^{U}\end{aligned} \tag{16}$$

Therefore, the decorrelation loss $\mathcal{L}_{dl}$ can be considered as a constant $\psi \ge 0$.

Therefore, the upper bound of $\mathcal{L}_{CCA}$ is a constant $C \in \mathbb{R}$, where $C \ge 0$. Combining each component's upper bound, the final upper bound is

$$\mathcal{L}_{\Theta}(\mathcal{M}) \le 2(\alpha m + (1-\alpha)v + \tau)\prod_{k=1}^{K}\|W_k\|_2 + C \tag{17}$$

where $\tau \le \prod_{k=1}^{K}\|W_k\|_2$.

REFERENCES

[1] 2023. DBLP dataset. https://dblp.uni-trier.de/db/

[2] Paolo Boldi, Francesco Bonchi, Aris Gionis, and Tamir Tassa. 2012. Injecting uncertainty in graphs for identity obfuscation. *arXiv preprint arXiv:1208.4145* (2012).

[3] OpenStreetMap Contributors. 2017. Roadnet Data. https://www.openstreetmap.org

[4] Herbert Federer. 2014. *Geometric measure theory.* Springer.

[5] Nevan J Krogan, Gerard Cagney, Haiyuan Yu, Gouqing Zhong, Xinghua Guo, Alexandr Ignatchenko, Joyce Li, Shuye Pu, Nira Datta, Aaron P Tikuisis, et al. 2006. Global landscape of protein complexes in the yeast Saccharomyces cerevisiae. *Nature* 440, 7084 (2006), 637–643.

[6] Jure Leskovec, Kevin J Lang, Anirban Dasgupta, and Michael W Mahoney. 2009. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics* 6, 1 (2009), 29–123.

[7] Chenhao Ma, Reynold Cheng, Laks VS Lakshmanan, Tobias Grubenmann, Yixiang Fang, and Xiaodong Li. 2019. Linc: a motif counting algorithm for uncertain graphs. *Proceedings of the VLDB Endowment* 13, 2 (2019), 155–168.

[8] Samuel Madden. 2004. Intel lab data. http://db.csail.mit.edu/labdata/labdata.html

[9] Panos Parchas, Nikolaos Papailiou, Dimitris Papadias, and Francesco Bonchi. 2018. Uncertain graph sparsification. *IEEE Transactions on Knowledge and Data Engineering* 30, 12 (2018), 2435–2449.

[10] Michalis Potamias, Francesco Bonchi, Aristides Gionis, and George Kollios. 2010. K-nearest neighbors in uncertain graphs. *Proceedings of the VLDB Endowment* 3, 1-2 (2010), 997–1008.

[11] Aladin Virmaux and Kevin Scaman. 2018. Lipschitz regularity of deep neural networks: analysis and efficient estimation. *Advances in Neural Information Processing Systems* 31 (2018).

[12] Hengrui Zhang, Qitian Wu, Junchi Yan, David Wipf, and Philip S Yu. 2021. From canonical correlation analysis to self-supervised graph neural networks. *Advances in Neural Information Processing Systems* 34 (2021), 76–89.