Flaky Performances When Pretraining on Relational Databases (Student Abstract)

Shengchao Liu[*1, 2], **David Vazquez**[3], **Jian Tang**[1, 4, 5], **Pierre-André Noël**[3]
[1] Mila, Québec AI Institute
[2] Université de Montréal
[3] ServiceNow Research
[4] HEC Montréal
[5] CIFAR AI Chair
liusheng@mila.quebec, pierre-andre.noel@servicenow.com

Abstract

We explore the downstream task performances for graph neural network (GNN) self-supervised learning (SSL) methods trained on subgraphs extracted from relational databases (RDBs). Intuitively, this joint use of SSL and GNNs should allow to leverage more of the available data, which could translate to better results. However, we found that naively porting contrastive SSL techniques can cause "negative transfer": linear evaluation on fixed representations from a pretrained model performs worse than on representations from the randomly-initialized model. Based on the conjecture that contrastive SSL conflicts with the message passing layers of the GNN, we propose InfoNode: a contrastive loss aiming to maximize the mutual information between a node's initial- and final-layer representation. The primary empirical results support our conjecture and the effectiveness of InfoNode.

Introduction

The success story of large language models hinges on self-supervised learning (SSL). Deep neural networks (DNNs) in other domains have similarly benefited from different SSL techniques, including some based on image augmentations.

Relational database (RDB) are typically modeled with fully-supervised, non-deep machine learning (ML): one first "flattens" the RDB to a single table, enabling ML models accepting "tabular data", a domain that has recently been called "the last *unconquered castle* for deep learning" (Kadra et al. 2021). Yet DNNs need not restrict themselves to tabular inputs: they may leverage more of the original RDB's structure, an hypothesis supported by graph neural networks (GNNs) work (Cvitkovic 2020). However, publications on deep graph-based models for RDB data remain rare: even with access to extra graph information, systematically beating tree models on RDBs flattened with deep feature synthesis (DFS) remains a challenge (Cvitkovic 2020).

SSL presents another opportunity for DNNs: we often have access to numerous unlabeled RDB entries in addition to the few labeled ones. However, under certain circumstances, pretraining on unlabeled data (followed by linear probing) can perform worse than an untrained model, including some seemingly "reasonable" choices of SSL strategy (details in Appendix[1]). Following these observations, we conjecture that the contrastive pretraining on RDB data is very sensitive to the view construction.

Contributions. To the best of our knowledge, we are the first to approach classification tasks on RDB using GNNs while leveraging unlabeled data using SSL pretraining. We empirically show that, while the use of SSL may confer some advantages for some datasets, SSL can actually lead to severe performance decrease, *i.e.*, negative transfer. We introduce InfoNode, which helps to certain extent.

Self-supervised Learning on RDB Graph

In this section, we discuss three SSL pretraining strategies—Generative, InfoNode and Hybrid—on the RDB graph. In addition, we also observe that existing graph contrastive SSL methods can bring in severe negative transfer issue.

Generative SSL. Denoising tasks are one of the most widely-used generative SSL methods. Here we mask-out a small fraction of the node attributes by replacing them by random values. Concretely, for each node i, a binary mask vector β of the same length as A_i is generated, a node j of the same type as i is randomly selected from the current batch, and the masked attributes are $A'_i = \beta \cdot A_i + (1-\beta) \cdot A_j$. The objective is then to recover the original attributes A from the noisy representation $\boldsymbol{h}'^T = \text{MPNN}(A', E)$. The loss $\mathcal{L}_G$ is a sum of mean squared errors for the continuous attributes and of cross entropies for the categorical ones.

Contrastive SSL: InfoNode. "Over-smoothing" is a well-known issue with GNNs: node-level representations may become indistinguishable and prediction performance may thus severely degrade as the number of layers increases. Conjecturing that it may be desirable for a node to "remember about itself", we introduce InfoNode: a node's initial ($\boldsymbol{h}_i^0$) and final ($\boldsymbol{h}_i^T$) representations become two views for a contrastive loss ($\boldsymbol{h}_g$ is the graph embedding):

$$\mathcal{L}_{\text{C-InfoNode}} = \mathbb{E}_{(i,g)\in\text{Pos}}\big[\sigma(f(\boldsymbol{h}_i^0, \boldsymbol{h}_i^T) + \sigma(f(\boldsymbol{h}_i^T, \boldsymbol{h}_g)\big] \quad (1)$$
$$+ \mathbb{E}_{(i',j')\in\text{Neg}}\big[1 - \sigma(f(\boldsymbol{h}_{i'}^0, \boldsymbol{h}_{j'}^T)\big] + \mathbb{E}_{(i',g')\in\text{Neg}}\big[1 - \sigma(f(\boldsymbol{h}_{i'}^T, \boldsymbol{h}_{g'})\big].$$

Hybrid objective. We follow Liu et al. (2022), where combining contrastive and generative SSL can augment the pretrained representation. Writing α_0 and α_1 the coefficients

[1] Appendix available at https://arxiv.org/abs/2211.05213.
[*] Work done during internship at ServiceNow Research.

SSL pretraining	Model	Acquire (160k)		Home Credit (307k)		KDD Cup (619k)	
		S=10%	S=100%	S=10%	S=100%	S=10%	S=100%
Untrained (random init)	GCN	54.36 ± 0.12	54.16 ± 0.22	52.00 ± 0.08	56.37 ± 1.72	51.78 ± 1.03	58.95 ± 0.45
	PNA	58.71 ± 0.73	61.68 ± 0.33	55.75 ± 1.96	62.76 ± 0.96	56.08 ± 2.43	62.05 ± 1.29
Generative	GCN	56.78 ± 0.08	58.19 ± 0.11	55.85 ± 0.00	62.52 ± 0.05	59.38 ± 0.05	64.38 ± 0.01
	PNA	64.85 ± 0.12	66.34 ± 0.06	$\mathbf{61.53 \pm 0.04}$	$\mathbf{67.43 \pm 0.04}$	$\mathbf{65.65 \pm 0.01}$	69.11 ± 0.06
InfoNode	GCN	53.76 ± 0.06	54.12 ± 0.09	54.78 ± 0.00	56.00 ± 0.04	52.69 ± 0.02	53.98 ± 0.08
	PNA	51.51 ± 0.33	51.64 ± 0.22	55.09 ± 0.49	55.80 ± 0.40	51.81 ± 0.42	53.12 ± 0.12
Hybrid	GCN	56.02 ± 0.15	58.19 ± 0.11	55.69 ± 0.02	59.79 ± 0.09	57.33 ± 0.01	60.21 ± 0.02
	PNA	$\mathbf{65.42 \pm 0.00}$	$\mathbf{66.60 \pm 0.03}$	59.35 ± 0.07	66.46 ± 0.15	64.52 ± 0.07	$\mathbf{70.24 \pm 0.02}$

Table 1: Main results with linear probing. In all cases, a linear classifier is trained on the representations of frozen models. For Untrained, the models are still in their randomly-initialized state. In the remaining rows, models are first pretrained with different SSL strategies before being frozen. Using InfoNode alone may cause performances to drop.

for the generative and contrastive objectives, the resulting objective function is:

$$\mathcal{L} = \alpha_0 \cdot \mathcal{L}_{\text{G}} + \alpha_1 \cdot \mathcal{L}_{\text{C-InfoNode}}. \tag{2}$$

Experiments and Discussion

Pipeline. We adopt the pretraining and linear-probing pipeline, *i.e.*, we will do SSL pretraining first, then we will fix the encoder and only fine-tune the prediction head. We adopt linear-probing because it can directly reflect the expressiveness of the pretrained model.

Datasets and evaluation. We consider the same 3 RDB datasets as in Cvitkovic (2020), all pre-processed with RDBToGraph. For these datasets, the predicted labels are binary and imbalanced, motivating the use of ROC-AUC. In addition, we use the whole training dataset for unsupervised pretraining, and then sample $S\%$ for downstream.

Backbone models and baselines. For the backbone GNNs (Kipf and Welling 2016), we consider GCN and PNA. The readout function is an attention module. For the pretraining methods, we first consider an untrained version (*i.e.*, without any pretraining). Then we consider the generative SSL, contrastive SSL, and the hybrid of the two.

Main results. Table 1 reports linear probing (LP) results as an indicator of the quality of the representations learned by different SSL strategies. Generative SSL shows quite consistent improvements. Interestingly, contrastive methods taken on their own perform rather poorly from this linear probing perspective. The learned representations are at best comparable to random representations, and in many cases are much worse ("negative transfer"). While not being particularly impressive, the hybrid SSL results do not show this counter-intuitive behavior. This generative/contrastive dichotomy is less visible in fine tuning (Appendix), possibly because the models are given the opportunity to "unlearn" bad representations. This observation also holds for the other contrastive pretraining methods on graph, yet our proposed InfoNode can alleviate the negative transfer issue better. Please see Appendix for more details.

Analysis. According to our hypothesis, leveraging unlabeled data with SSL should typically improve downstream task performances. Of course, we were aware that there is no free lunch: due to its inductive biases, a model may be good for some tasks and bad for others. However, we believe that our results are not just random edge cases, but instead reveal a more systematic SSL failure mode. *E.g.*, similar phenomenon has been observed in molecular graphs (Liu, Guo, and Tang 2022). In particular, we posit that RDB data distributions contain *traps*—"interesting-looking noise"—that some SSL strategies may "fall for", and that "better" models may be more prone to these traps. As an illustration of how such traps may exist, consider a single-table RDB with 3 non-label columns—so a graph made of an isolated node with 3 properties—and suppose that its probability distribution factorizes as $\mathrm{P}(A) = \mathrm{P}(A_{00})\,\mathrm{P}(A_{01}, A_{02})$. Given unlabeled data samples A, the "best" that any SSL strategy could do is to learn $\mathrm{P}(A_{00})$ and $\mathrm{P}(A_{01}, A_{02})$. The ability to uncover the presence of mutual information $\mathrm{I}(A_{01}; A_{02})$ between the corresponding datum is one of the characteristics typically associated with "good" SSL models, *but such models may neglect A_{00}, and A_{00} may be all that matters for some downstream tasks*. More details are in Appendix.

Conclusion and Future Direction. In this work, we propose a novel contrastive pretraining method, InfoNode, to alleviate the inherent issue of GNN. Primary experiments motivate further investigations as to the mechanisms involved.

References

Cvitkovic, M. 2020. Supervised Learning on Relational Databases with Graph Neural Networks. *arXiv preprint arXiv:2002.02046*.

Kadra, A.; Lindauer, M.; Hutter, F.; and Grabocka, J. 2021. Well-tuned Simple Nets Excel on Tabular Datasets. *Advances in Neural Information Processing Systems*, 34.

Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Liu, S.; Guo, H.; and Tang, J. 2022. Molecular geometry pretraining with se (3)-invariant denoising distance matching. *arXiv preprint arXiv:2206.13602*.

Liu, S.; Wang, H.; Liu, W.; Lasenby, J.; Guo, H.; and Tang, J. 2022. Pre-training Molecular Graph Representation with 3D Geometry. In *International Conference on Learning Representations*.