On Neural Scaling Laws on Graphs

Jingzhe Liu
Michigan State University
liujin33@msu.edu

Haitao Mao
Michigan State University
haitaoma@msu.edu

Zhikai Chen
Michigan State University
chenzh85@msu.edu

Tong Zhao
Snap Inc
tzhao@snap.com

Neil Shah
Snap Inc
nshah@snap.com

Jiliang Tang
Michigan State University
tangjili@msu.edu

Abstract

Deep graph models (e.g., graph neural networks and graph transformers) have become important techniques for leveraging knowledge across various types of graphs. Yet, the neural scaling laws on graphs, i.e., how the performance of deep graph models changes with model and dataset sizes, have not been systematically investigated, casting doubts on the feasibility of achieving large graph models. To fill this gap, we benchmark many graph datasets from different tasks and make an attempt to establish the neural scaling laws on graphs from both model and data perspectives. The model size we investigated is up to 100 million parameters, and the dataset size investigated is up to 50 million samples. We first verify the validity of such laws on graphs, establishing proper formulations to describe the scaling behaviors. For model scaling, we identify that despite the parameter numbers, the model depth also plays an important role in affecting the model scaling behaviors, which differs from observations in other domains such as CV and NLP. For data scaling, we suggest that the number of graphs can not effectively measure the graph data volume in scaling law since the sizes of different graphs are highly irregular. Instead, we reform the data scaling law with the number of nodes or edges as the metric to address the irregular graph sizes. We further demonstrate that the reformed law offers a unified view of the data scaling behaviors for various fundamental graph tasks including node classification, link prediction, and graph classification. This work provides valuable insights into neural scaling laws on graphs, which can serve as an important tool for collecting new graph data and developing large graph models.

1 Introduction

The neural scaling law [1–3] serves as an important tool in the development of large models. It quantitatively describes how model performance grows with the scale of training, which have two basic forms: data scaling law (how the performance changes with training dataset size) and model scaling law (how the performance changes with model size). Neural scaling laws guide the construction of large models from the following perspectives: (1) They enable us to predict model performance on large-scale training by extrapolating from small-scale runs with lower computational budgets. For instance, the performance of GPT-4 on coding problems fits accurately with the prediction of the scaling law [4]. (2) They enable us to identify better model architectures through small-scale training, typically selecting those with the largest performance gains for a given computational overhead. (3) They serves as the guidance for resource allocation, i.e., they could tell us how to distribute the cost of data collection and computation for the given model to reach the expected performance. Such laws have played an instrumental role in the recent successes of large models in natural language processing (NLP) and computer vision (CV) domains [5–9].

J. Liu et al., On Neural Scaling Laws on Graphs. *Proceedings of the Third Learning on Graphs Conference (LoG 2024)*, PMLR 269, Virtual Event, November 26–29, 2024.

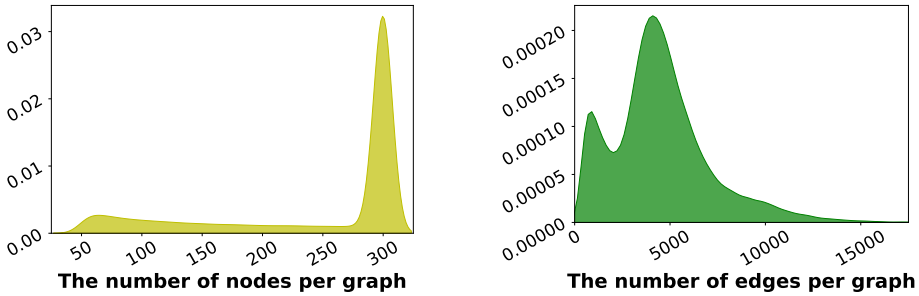


Figure 1: The density distribution of the number of nodes and edges in OGBG-PPA dataset.

Despite the success of foundation models in NLP and CV, there is no existing effort investigating the neural scaling law in the graph machine learning domain. As an initial exploration, we found that such an investigation would meet unique challenges given the irregularity and diversity of graphs. Specifically, both the number of nodes and edges of a graph can vary dramatically, different from the fixed-structure figures or texts in CV and NLP. As an example shown in Figure 1, in the OGBG-PPA dataset of $158k$ graphs [10], the number of nodes in each graph ranges from 25 to 325, and the number of edges from 54 to $18,096$, indicating significant variability across the dataset. The variability in size suggests that the current metric (the number of graph samples) insufficiently represents the data volume, hinting at the need for more proper data metrics of the graph scaling laws. Moreover, the irregularity of graphs requires the deep graph models to introduce the message aggregation mechanisms. Hence, besides the model weights considered in the existing scaling law, deep graph models also need to consider the influence of the non-parametric aggregations, i.e., whether graph models with different numbers of aggregation layers could exhibit distinct scaling behaviors.

In this work, we make a first attempt to address these unique challenges and develop the neural scaling laws on graphs to become a tool for building large graph models. Our investigation comes across a large range of domains and tasks in graph representation learning. Experimental observations reveal neural scaling laws across different models and dataset sizes for graph learning tasks.

Our contributions can be summarized as follows:

- We start our investigation with verifying the general forms of graph scaling laws. Specifically, we develop math formulas for the relation between model/dataset size and the model performance. Then we verify their validity with empirical studies across different datasets and models.
- We find that deep graph models with different numbers of layers will exhibit different model scaling behaviors, contrasting with the conclusions in CV and NLP domains. We then look deeper into the phenomenon and propose that the non-parametric aggregation operation could be one of the possible reasons.
- Regarding the size variability problem, we reveal that the number of nodes or edges is a better metric than the number of graphs when applying the data scaling law in graph classification tasks. To support this argument, we demonstrate that models trained on datasets with identical graph counts but different total numbers of edges exhibit distinctive scaling behaviors.
- We further extend the data scaling law to node classification and link prediction tasks, demonstrating the generality of using the number of nodes or edges as the data metric.

Orgnizations. The main focus of this work is on the two basic forms of neural scaling laws for the graph domain, i.e., the modeling scaling law and the data scaling law. We tend to address the challenges caused by the irregularity of graph data when establishing the scaling laws. Our investigations focus on the well-studied supervised graph learning while leaving the under-studied unsupervised learning tasks [11] as one future work. The following sections will be arranged as follows: In Section 2, we briefly introduce the neural scaling laws and the related works. In Section 3, we introduce the basic settings of our experiments. In Section 4, we verify the validity of the basic forms of neural scaling laws on the graphs. In Section 5, we reveal that model depth can affect the model scaling behaviors, which is a unique phenomenon in graph domains. In Section 6, we propose the number of edges is a suitable data metric when applying the data scaling law on graphs. To support the argument, we exhibit that models would have distinctions in scaling behavior when trained on datasets with different numbers of edges. Furthermore, we present that using the number

of edges as the data metric could provide a unified data scaling law for various graph learning tasks, e.g., node classification, link prediction, and graph property prediction.

2 Related Work

Functional forms of neural scaling laws. Neural scaling laws give the quantitative relationship between the model performance and three main factors: the number of model parameters $\mathbf{N}$, the size of the training dataset $\mathbf{D}$, and the amount of computation $\mathbf{C}$. The relationship is generally described as a power-law, which has widely been applied in the domains of image classification [8, 12–14], generative language modeling [2, 13, 15], and natural language translation [16–19].

Using the test error ϵ as a metric for model performance, Hestness et al [12] propose the basic functional form of neural scaling laws as follows:

$$\epsilon = \mathbf{a}\mathbf{X}^{-\mathbf{b}} + \epsilon_\infty \tag{1}$$

The variable $\mathbf{X}$ represents any of the parameters $\mathbf{N}$, $\mathbf{D}$, or $\mathbf{C}$. The positive constants $\mathbf{a}$ and $\mathbf{b}$ govern the rate of error reduction as a function of $\mathbf{X}$, while $\epsilon_\infty > 0$ specifies the minimum achievable error. Equation 1 is appropriate for regression tasks but is ill-defined for classification tasks, since the error rate $\epsilon_\mathbf{r}$ remains finite for models with zero training. To fix this problem, Zhai et al [8] proposes a modified function:

$$\epsilon_\mathbf{r} = \mathbf{a}(\mathbf{X} + \mathbf{c})^{-\mathbf{b}} + \epsilon_\infty \tag{2}$$

where a positive shift coefficient $\mathbf{c}$ is added to avoid having infinite $\epsilon_\mathbf{r}$ when $\mathbf{X}$ equals to zero. In this work, we will base our following investigations of graph classification tasks on this equation.

The relationship between $\mathbf{N}$, $\mathbf{D}$ and overfitting. The negative exponent $-\mathbf{b}$ in Equation 2 indicates that the performance gain will be diminishing if only one of the parameters $\mathbf{N}$ and $\mathbf{D}$ is increased. To break this bottleneck, the dataset size and the model size need to be enlarged together. Specifically, Kaplan et al [2] find that the model size and dataset size should increase together as $\mathbf{D} \sim \mathbf{N}^\alpha$ $(0 < \alpha < 1)$ to attain optimal return for compute budgets. It is also observed that overfitting will likely occur if $\mathbf{N}$ surpasses $\mathbf{D}^{\mathbf{1}/\alpha}$ significantly, which leads to negative performance gain.

Graph learning and scaling. The graph is capable of representing non-Euclidean data with irregular and flexible structures, e.g., social networks [20], molecules [21], and protein networks [10]. Despite the popularity of graph machine learning, there are seldom large graph models. Notably, most deep GNNs remain small parameter size but more aggregation layers. Moreover, the neural scaling laws on graphs remain underexplored. Some works have looked into the neural scaling laws for molecule representation learning [22, 23], but the exploration is limited to the data and model types. It is unclear how the conclusion can be generalized to other graphs. To the best of our knowledge, we are the first to investigate neural scaling laws for the general graph domain.

3 Preliminaries and Experiment Setups

To ensure the applicability of neural scaling laws in the graph domain, our experiments are comprehensive to cover different types of models and graphs from various domains, e.g., molecule, biology, and academic. We investigate two major types of deep graph learning models, i.e., graph neural networks (GNNs) and graph transformers. We explore GCN [24] and GIN [25] for GNNs, and SAT [26], and GraphGPS [27] for graph transformers. Our datasets [28–30] cover the the most common graph domains including social networks (REDDIT-MULTI-12K, OGBL-COLLAB), molecules (OGBG-MOLHIV,OGBG-MOLPCBA,PCQM4MV2), protein networks (OGBG-PPA), and citation networks (OGBN-ARXIV). Also, the chosen datasets are relatively large-scale ones among the academic datasets, ensuring valid observation of scaling behaviors. Moreover, our experiments are conducted on three common graph tasks, i.e. graph property prediction, node classification, and link prediction, though graph property prediction is the focus. Notably, accuracy rather than error rate is adopted in graph classification tasks as the default metric. To ensure a consistent observation, we switch the error rate ϵ_r in Equation 2 to accuracy via applying the relation *Accuracy* $= 1 -$ *ErrorRate* and obtain the following equation:

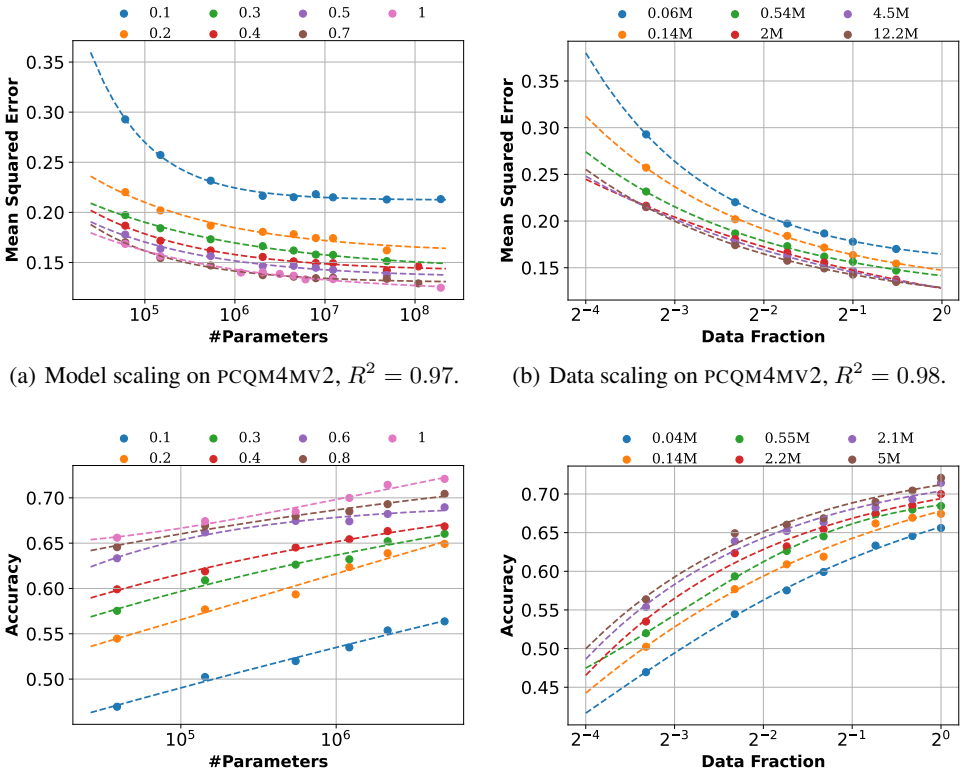
$$\mathbf{s} = -\mathbf{a}(\mathbf{X} + \mathbf{c})^{-\mathbf{b}} + \mathbf{s}_\infty \tag{3}$$

where $\mathbf{s}$ is the test score corresponding to specific metrics and $\mathbf{s}_\infty = \mathbf{1} - \epsilon_\infty$. Equation 3 will be used in the following sections to fit the scaling law on classification tasks. The details of datasets and experimental settings can be found in Appendix A.

4 Validity of Basic Scaling Laws on Graphs

We start our analysis by examining the basic forms of neural scaling laws in the graph domain. In other words, we aim to explore whether the model performance has a power-law relationship with the dataset size and model size. Hence, we first evaluate the performance of the models across various sizes and training set sizes. Since the impact model depth is unclear yet, we vary the model size by increasing the model width while fixing the model depth, and the effect of model depth will be further explored in Section 5. To obtain subsets with different sizes, we randomly subsample the original training set by different sampling ratios. Then, we fit the scaling law Equation 1 and 3 to the empirical results on different model scales using the least square method. To measure the fitting quality, the coefficient of determination R^2 is adopted as a metric.

In this subsection, we focus on GIN and leave the results with similar observations on graph transformer in Section 5 on GCN in Appendix B. Our experiments are conducted on two large graph property prediction datasets: PCQM4MV2 and OGBG-PPA.



(a) Model scaling on PCQM4MV2, $R^2 = 0.97$.

(b) Data scaling on PCQM4MV2, $R^2 = 0.98$.

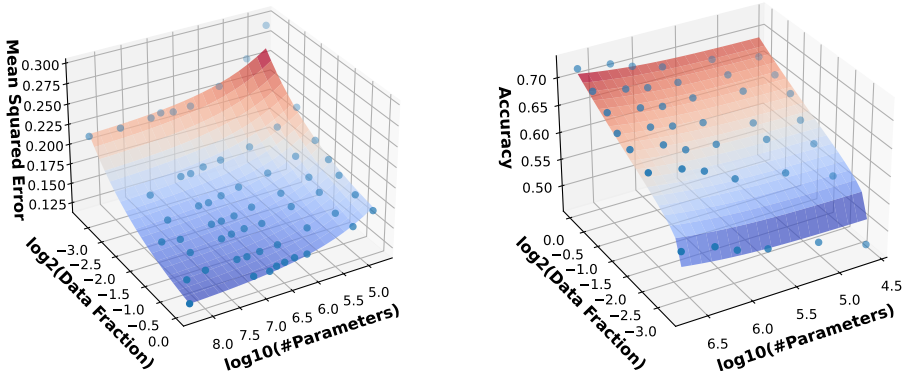
(c) Model scaling on OGBG-PPA, $R^2 = 0.98$.

(d) Data scaling on OGBG-PPA, $R^2 = 0.98$.

Figure 2: The model and data scaling behaviors of GIN on both PCQM4MV2 and OGBG-PPA. For model scaling, the color corresponds to the training data fraction; for data scaling, the color corresponds to the number of model parameters. The lower the mean square error is, the better the model performance is. The high R^2 values indicate that the scaling law curves fit well with the empirical values.

We conduct experiments for both model scaling and data scaling behaviors on the graph regression dataset PCQM4MV2. For regression tasks, the performance is measured by mean square error (MSE), hence we use Equation 1 to fit the empirical MSEs. For model scaling, we fix the number of layers for the model to 6 and vary the model width from 64 to $1,024$. The experiments are repeated for different data fractions chosen from $[0.1, 0.2, 0.3, 0.4, 0.5, 0.7, 1]$. The results, shown in Figure 2(a), depict each model's MSE as dots, with dashed lines representing fitted scaling law curves and colors indicating specific training data fractions. We make the following observations from Figure 2(a): (1) The MSE tends to decrease when increasing the model scale and fixing the dataset scale. (2)

The MSE gradually decreases and approaches a lower bound ϵ asymptotically as the model becomes larger. The scaling curves corresponding to larger dataset sizes would have smaller ϵ, i.e., higher upper bound of the model performance. The observations in Figure 2(b) exhibit similar patterns when scaling up data size while fixing the model size. Generally speaking, both model scaling curves in Figure 2(a) and data scaling curves in Figure 2(b) fit well to empirical values with high R^2s, indicating that Equation 1 could be leveraged to describe the neural scaling laws for graph regression tasks.



(a) Scaling behaviors on PCQM4MV2, $R^2 = 0.98$. (b) Scaling behaviors on OGBG-PPA, $R^2 = 0.86$.

Figure 3: The comprehensive scaling behaviors of GIN on PCQM4MV2 and OGBG-PPA datasets. For PCQM4MV2, the color corresponds to the value of mean square error; for OGBG-PPA, the color corresponds to accuracy. The surfaces predicted by the neural scaling law fit well with the empirical values on both datasets, as indicated by the high R^2 values.

More scaling behaviors on the OGBG-PPA dataset are illustrated in Figure 2(c) and 2(d). The way of varying dataset sizes and model sizes is similar to that of PCQM4MV2. However, since the task on OGBG-PPA is graph classification, we use accuracy as the performance metric and apply Equation 3 to fit the scaling behaviors. Observations from Figure 2(c) and 2(d) align with those from Figure 2(a) and 2(b), suggesting that Equation 3 can describe the scaling behavior for graph classification tasks.

On the basis of the validity of data scaling law and model scaling law individually, we further show that the two formulations can be combined into a comprehensive form, similar to [1]. For regression, the form is:

$$\epsilon = \mathbf{a_1 D^{-b_2}} + \mathbf{a_2 N^{-b_2}} + \epsilon_\infty \tag{4}$$

For classification tasks, the comprehensive form is:

$$\mathbf{s} = -\mathbf{a_1}(\mathbf{D} + \mathbf{c_1})^{-\mathbf{b_1}} - \mathbf{a_2}(\mathbf{N} + \mathbf{c_2})^{-\mathbf{b_2}} + \mathbf{s}_\infty \tag{5}$$

We verify the comprehensive forms by fitting the empirical values to Equation 4 and 5. The experimental results are shown in Figure 3, where the high R^2 values suggest that the model scaling and data scaling laws can be well unified by Equation 4 and 5.

Observation 1. ***The model and data scaling behaviors in the graph domain can be described by neural scaling laws.***

The above observations suggest that the scaling behaviors for graph property prediction tasks follow neural scaling laws and can be correctly predicted by Equation 1 and 3. Thus, it is possible to attain expected performance on graph tasks by enlarging the model and dataset size, exhibiting the potential of building large graph models. Despite the general success, there are still some unique chanllenges for scaling laws in the graph domain. We will discuss the challenges for the model scaling in Section 5, and those for the data scaling in Section 6.

5 The Effect of Model Depth on Graph Scaling Laws

A unique question in the graph domain is whether the model depth would affect the model scaling law, or more specifically, the scaling law coefficients in Equation 3. Previous research [2] in the NLP

domain finds that the model depth does not have effects on the scaling coefficients, i.e., the scaling curves for transformers of different numbers of model layers will overlap. Therefore, conducting preliminary experiments to determine the optimal number of layers for large language models (LLMs) is not required, which effectively narrows the hyperparameter search space.

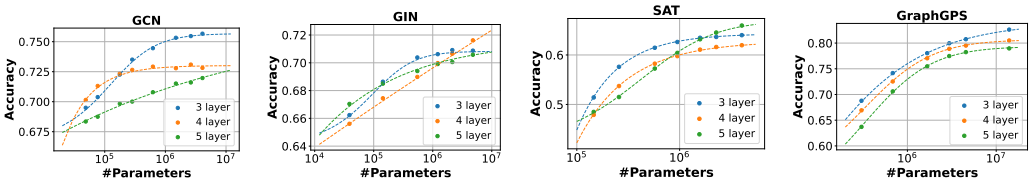


Figure 4: Model scaling behaviors of GNNs and graph transformers with varying model depths. The color corresponds to the model layer. All the models exhibit distinct scaling behaviors with different model depths.

Nonetheless, the deep graph models have different architectures, i.e., non-parametric aggregation layer, from vanilla transformers, which induces unique problems in graph learning such as over-smoothing [31] and over-squashing [32]. Thus, it is desired to study whether the conclusion from the NLP domain still holds on graphs.

To ensure the generalization of our investigations, we choose two representative GNNs (GCN & GIN) and two powerful graph transformers (SAT & GraphGPS). For these models, we choose the layer number in $[3, 4, 5]$ and fix it but change the model width to attain varying model sizes. All the models are trained on the whole OGBG-PPA training set and then evaluated for model performance. The model scaling curves of models with different model depths are presented in Figure 4. Observations can be made as follows: (1) The scaling behaviors of models with different numbers of layers are distinct. For each model, the shape of the scaling law curves would change when the model depth changes, exhibiting different increasing speeds and upper bounds for performance. (2) The optimal number of layers for different models could be different. For GCN and GraphGPS, models with 3 layers have the best performance, but for GIN the optimal layer number is 4, and for SAT is 5. Similar observations on more datasets can be found in Appendix E.

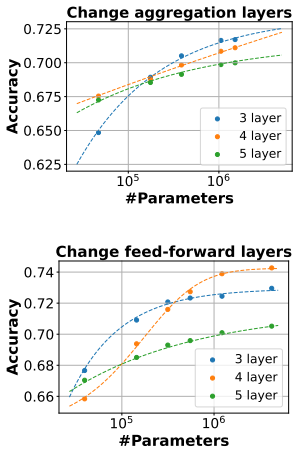


Figure 5: Model scaling behaviors when fixing the number of aggregation layers or feed-forward layers.

Observation 2. ***Deep graph models with varying depths will have different model scaling behaviors.***

Current observations support that there is no one-size-fits-all model depth for existing graph learning models. The optimal number of model layers could be task-specific and model-specific. We emphasize the necessity of preliminary experiments for the optimal number of model layers before scaling up the model parameters. Further, we look deep into the relationship between the non-parametric aggregation operation and the layer difference of graph model scaling. To achieve this goal, we first decouple the GIN layer into one feed-forward layer and one non-parametric aggregation layer, then repeat the above experiments by varying the number of feed-forward or aggregation layers while fixing the other to be 5. The results are shown in Figure 5. We observe that no matter whether fixing the aggregation layer or feed-forward layer, there will always be layer differences in model scaling. These observations suggest that the non-parametric aggregation could be one of the reasons for the layer difference. As for the observations when increasing the feed-forward layers, we speculate the local reception field of GNN (different from the global attention of vallina transformer) to be the reason. Notably, we do not have any principle guidance helping to automatically select the best model layer so far and we still need to select the optimal layer number for specific graph learning tasks. Thus, building a new model whose model scaling behaviors are agnostic to the number of layers could be beneficial to the development of large graph models.

6 Proper Data Metric of Graph Scaling Laws

In this section, we look into the data scaling law on graphs. Specifically, we re-examine data metrics in previous studies and reveal the disadvantage of the number of graphs as the data metric in the data scaling law. For instance, a small molecule graph could have a large size difference compared with a social network. Instead, we identify the total number of nodes or edges in the training set as a better metric with experimental evidence.

In previous studies, the data metric in scaling laws is the number of graph samples, i.e., how many graphs in the whole training set [1, 22]. Nonetheless, using the number of graphs as the metric neglects the size variability of graphs. For instance, a graph with 10 edges and a graph with $1,000$ edges are treated equivalently when the number of graphs is the metric, even though they could differ remarkably in training costs and impacts on model performance improvement. Empirically, we conduct two controlled experiments to show that the number of nodes or edges is the better data metric in general.

Experiment 1: From the data scaling perspective. In this experiment, we show that the number of nodes (edges) as the data metric can provide a better fitting of scaling laws than the number of graphs. To highlight the contrast, the training set is scaled up in the following way: each time we add a subset with the same number of graphs but an increasing number of total nodes/edges. However, the graph size distribution often varies among classes of real-world datasets. To eliminate the bias, we adopt the setup in [33] and construct a synthetic dataset with a motif recognition task. Notably, in the setup [33], the synthetic graphs are generated as Barabasi Albert graphs, whose nodes are connected with a fixed number of edges. Thus, we can view the numbers of nodes and edges as equivalent in this setting. Specifically, the dataset includes three classes, i.e., three motifs (Cycle, Grid, House) to recognize. We generate 8 subgroups for each class, where each subgroup contains 100 graphs with the same number of nodes i ($i \in [25, 50, 100, 200, 400, 800, 1600, 3200]$). Each time to increase the training set size, we add the subgroups of three classes with the same number of nodes i (i increases monotonically from 25 to 3200). Then we can ensure that the total number of graphs from each class is balanced. After recording the model performance of varying training set sizes, we fit the data scaling law to the empirical values with the number of nodes (edges) as the data metric repectively. The results of the number of edges are shown in Figure 6. Results with the number of nodes as the metric can be found in Appendix G. From the figure, we observe that the number of nodes as the data metric gives a more accurate fitting with a higher R^2 value, which supports our argument.

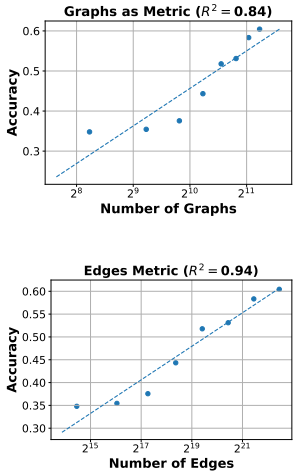


Figure 6: Comparison between the number of graphs and edges as data metric

Experiment 2: From the model scaling perspective. For real-world datasets, we contrast the two metrics by comparing the model scaling curves of the same model on two training sets with the same number of graphs but different number of nodes/edges. Since this study focused on in-domain graph learning, the graph samples tend to have similar average degrees, as shown in Figure 7. The degree distributions of other graph datasets have the same trend, which can be found in Appendix F. Hence, we could treat the number of nodes and edges as nearly equivalent metrics. We discuss the situations with the number of edges as the metric in this subsection and results with the number of nodes can be found in Appendix G. We first divide the whole training sets of one dataset into two subsets as follows: (1) We first arrange all graph samples of class i into a list $\mathcal{G}_i$ in an ascending order of their numbers of edges. (2) We choose the first half of all $\mathcal{G}_i$ as the "simple" training subset, and then the second half as the "complex" subset. Thus, these two subsets have the same number of graphs while the graphs in the complex subset have more edges. The validation and test sets remain as the original. Then we train GIN of varying sizes on these two subsets and report the performance on OGBG-MOLHIV in Figure 8.

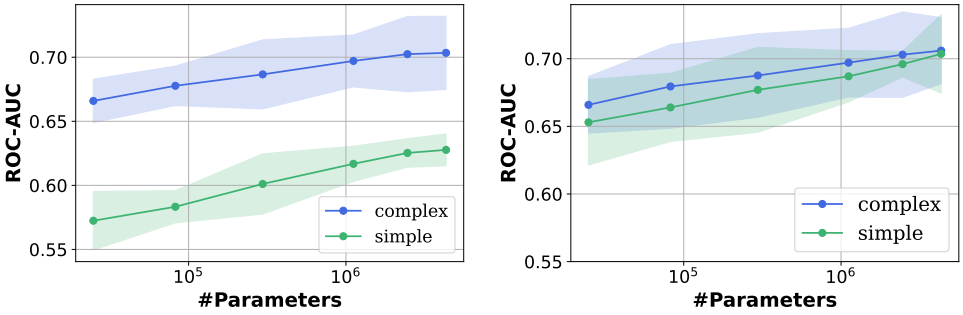


Figure 8: Model scaling curves of GIN on two training subsets of different sizes. (a) Two subsets have the same total number of graphs but different total number of edges. (b) Two subsets have the same total number of edges but different total number of graphs. The shade corresponds to the performance variance of repeated experiments.

If the number of graphs can accurately metric the data volume, the curves of simple and complex subsets in Figure 8(a) should have small differences or even overlap with each other. However, their performance curves exhibit considerable differences, indicating that the two subsets contain different amounts of data. For comparison, we attain the two subsets in another way: the simple subset has the first m samples in $\mathcal{G}_i$, and the complex subset has the remaining n samples, where the two subsets contain the same number of edges. It is easy to validate that $m > n$. Then we train models of varying sizes on these two subsets and report their performance on OGBG-MOLHIV in Figure 8(a).

Compared to Figure 8(a), the curves in Figure 8(b) have much smaller differences, which indicates that the total number of edges could be a better metric than the number of graphs.

Observation 3. ***The number of edges is a better data metric for data scaling law compared to the number of graphs.***

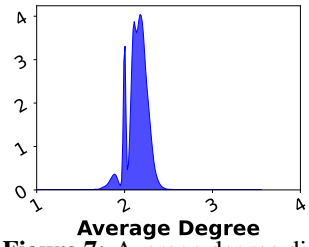


Figure 7: Average degree distribution of OGBG-MOLHIV dataset.

In addition, using the total number of edges as the data metric in scaling laws has two more advantages: (1) Knowing the GNN architecture and the number of edges in a dataset, the compute cost of training can be directly calculated, which enables us to build connections between data scaling law and compute scaling law as in [2]. The detailed explanation is in Appendix I. (2) Furthermore, using the number of edges as the data metric potentially provides a unified way to describe the data scaling behaviors of node classification and link prediction tasks, which will be thoroughly studied in the following section.

7 Results on Node Classification and Link Prediction

Furthermore, the data scaling law with the number of nodes (edges) as the data metric can be naturally extended to node classification and link prediction tasks, potentially providing a unified view of data scaling behaviors in the graph domain. There is only one graph in most of the node classification and link prediction datasets, thus it is infeasible to use the number of graphs to measure the training data volume. Nonetheless, the total number of nodes (edges) is able to serve as the data metric for these tasks. Moreover, with the total number of nodes (edges) as the data metric, we can potentially attain a unified data scaling law on graphs by directly extending Equation 3 to node classification and link prediction tasks. To verify this, we conduct experiments on the data scaling behaviors on both node classification and link prediction tasks. Here we only provide results with the number of edges as metric, for the results with the number of edges as metric, please refer to Appendix H. For node classification, we train a 3-layer GraphSAGE [34] on OGBN-ARXIV, OGBN-PRODUCTS and OGBN-PROTEINS, where the hidden vector size is set to 128. For link prediction, we train a 3-layer SEAL [35] on OGBL-COLLAB, OGBL-PPA and OGBL-CITATION2, where the model takes one-hop neighbors of the

target node pairs as input and its hidden vector size is set to 256. We take the total number of edges on which models are trained as the training set size for both tasks. The results are demonstrated in Figure 9. Similar to graph classification tasks, Equation 3 fits well to the data scaling behaviors, as evidenced by the high R^2 values.

Observation 5. ***Node classification, link prediction, and graph classification tasks could follow the same form of data scaling law with the number of nodes (edges) as data metric.***

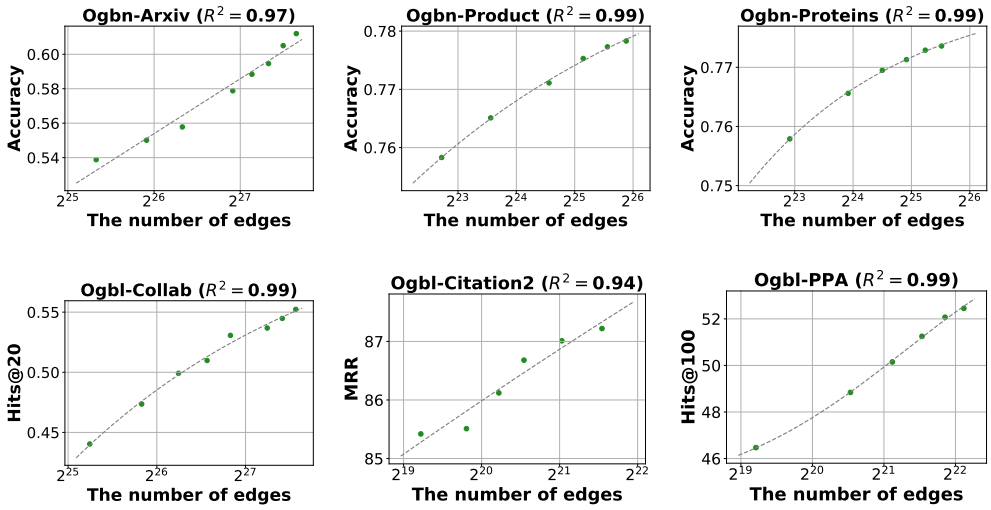


Figure 9: Both node classification and link prediction follow the data scaling law with the total edge number as the data metric.

8 Conclusions

In this work, we take the first step to investigate neural scaling laws on graphs. In particular, we focus on two basic forms of neural scaling laws: the model scaling law and the data scaling law. We first verify the formulations that can describe both scaling behaviors in the graph domain. For the model scaling law, we identify and investigate some graph-specific phenomena, revealing different observations from CV and NLP domains. For the data scaling law, we reform it by adopting the number of edges as the data metric and extend the reformed version to node classification and link prediction tasks. With more attention has been attracted to the graph foundation models [36–39], our findings shed light on the scaling behaviors of deep graph models and can provide valuable instructions for developing large graph models.

Limitations. Next, we discuss the limitations of this work, which could be further explored as future works. (1) There is no large graph dataset as large as CV or NLP datasets. We conduct our experiments on the largest graph datasets that we can attain in academics. Even though, this initial study is still meaningful. Only after the data scaling law on graphs has been verified, more efforts would be made to construct larger graph datasets, and the experience from this work could also be used to guide the process. For instance, Tang et al [40] adapt the data scaling scaling perspective and improve the crosse-domain graph learning performance by generating synthetic graph data for training by diffusion. (2) Since there has not been a well-established pretraining task for graph learning, we limit our investigation to the supervised learning on graphs. (3) In our exploration, we assume that all node (edge) features are in a unified feature space. We do not consider heterogeneous feature problems in this work.

References

[1] Jonathan S Rosenfeld, Amir Rosenfeld, Yonatan Belinkov, and Nir Shavit. A constructive prediction of the generalization error across scales. *arXiv preprint arXiv:1909.12673*, 2019. 1, 5, 7

[2] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. 3, 5, 8, 15, 21

[3] Samira Abnar, Mostafa Dehghani, Behnam Neyshabur, and Hanie Sedghi. Exploring the limits of large scale pre-training. *arXiv preprint arXiv:2110.02095*, 2021. 1

[4] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1

[6] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[7] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240): 1–113, 2023.

[8] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12104–12113, June 2022. 3

[9] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. In *International Conference on Machine Learning*, pages 7480–7512. PMLR, 2023. 1

[10] Damian Szklarczyk, Annika L Gable, David Lyon, Alexander Junge, Stefan Wyder, Jaime Huerta-Cepas, Milan Simonovic, Nadezhda T Doncheva, John H Morris, Peer Bork, et al. String v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic acids research*, 47(D1):D607–D613, 2019. 2, 3

[11] Ruoxi Sun, Hanjun Dai, and Adams Wei Yu. Does gnn pretraining help molecular representation? *Advances in Neural Information Processing Systems*, 35:12096–12109, 2022. 2

[12] Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017. 3

[13] Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B Brown, Prafulla Dhariwal, Scott Gray, et al. Scaling laws for autoregressive generative modeling. *arXiv preprint arXiv:2010.14701*, 2020. 3

[14] Utkarsh Sharma and Jared Kaplan. Scaling laws from the data manifold dimension. *The Journal of Machine Learning Research*, 23(1):343–376, 2022. 3

[15] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022. 3, 21

[16] Mitchell A Gordon, Kevin Duh, and Jared Kaplan. Data and parameter scaling laws for neural machine translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5915–5922, 2021. 3

[17] Behrooz Ghorbani, Orhan Firat, Markus Freitag, Ankur Bapna, Maxim Krikun, Xavier Garcia, Ciprian Chelba, and Colin Cherry. Scaling laws for neural machine translation. *arXiv preprint arXiv:2109.07740*, 2021.

[18] Yamini Bansal, Behrooz Ghorbani, Ankush Garg, Biao Zhang, Colin Cherry, Behnam Neyshabur, and Orhan Firat. Data scaling laws in NMT: The effect of noise and architecture. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*,

volume 162 of *Proceedings of Machine Learning Research*, pages 1466–1482. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/bansal22b.html.

[19] Patrick Fernandes, Behrooz Ghorbani, Xavier Garcia, Markus Freitag, and Orhan Firat. Scaling laws for multilingual neural machine translation. *arXiv preprint arXiv:2302.09650*, 2023. 3

[20] Pinar Yanardag and SVN Vishwanathan. Deep graph kernels. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1365–1374, 2015. 3

[21] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017. 3

[22] Dingshuo Chen, Yanqiao Zhu, Jieyu Zhang, Yuanqi Du, Zhixun Li, Qiang Liu, Shu Wu, and Liang Wang. Uncovering neural scaling laws in molecular representation learning. *arXiv preprint arXiv:2309.15123*, 2023. 3, 7

[23] Maciej Sypetkowski, Frederik Wenkel, Farimah Poursafaei, Nia Dickson, Karush Suri, Philip Fradkin, and Dominique Beaini. On the scalability of gnns for molecular graphs. *arXiv preprint arXiv:2404.11568*, 2024. 3

[24] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2016. 3

[25] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2018. 3

[26] Dexiong Chen, Leslie O'Bray, and Karsten Borgwardt. Structure-aware transformer for graph representation learning. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 3469–3489. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/chen22r.html. 3

[27] Ladislav Rampášek, Michael Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a general, powerful, scalable graph transformer. *Advances in Neural Information Processing Systems*, 35:14501–14515, 2022. 3

[28] Christopher Morris, Nils M Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. Tudataset: A collection of benchmark datasets for learning with graphs. *arXiv preprint arXiv:2007.08663*, 2020. 3

[29] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. *Advances in neural information processing systems*, 33:22118–22133, 2020.

[30] Weihua Hu, Matthias Fey, Hongyu Ren, Maho Nakata, Yuxiao Dong, and Jure Leskovec. Ogb-lsc: A large-scale challenge for machine learning on graphs. *arXiv preprint arXiv:2103.09430*, 2021. 3

[31] T Konstantin Rusch, Michael M Bronstein, and Siddhartha Mishra. A survey on oversmoothing in graph neural networks. *arXiv preprint arXiv:2303.10993*, 2023. 6

[32] Francesco Di Giovanni, T Konstantin Rusch, Michael M Bronstein, Andreea Deac, Marc Lackenby, Siddhartha Mishra, and Petar Veličković. How does over-squashing affect the power of gnns? *arXiv preprint arXiv:2306.03589*, 2023. 6

[33] Zhitao Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. Gnnexplainer: Generating explanations for graph neural networks. *Advances in neural information processing systems*, 32, 2019. 7

[34] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017. 8

[35] Muhan Zhang, Pan Li, Yinglong Xia, Kai Wang, and Long Jin. Labeling trick: A theory of using graph neural networks for multi-node representation learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 9061–9073. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/4be49c79f233b4f4070794825c323733-Paper.pdf. 8

[36] Haitao Mao, Zhikai Chen, Wenzhuo Tang, Jianan Zhao, Yao Ma, Tong Zhao, Neil Shah, Michael Galkin, and Jiliang Tang. Graph foundation models. *arXiv preprint arXiv:2402.02216*, 2024. 9

[37] Hao Liu, Jiarui Feng, Lecheng Kong, Ningyue Liang, Dacheng Tao, Yixin Chen, and Muhan Zhang. One for all: Towards training one graph model for all classification tasks. *arXiv preprint arXiv:2310.00149*, 2023.

[38] Qian Huang, Hongyu Ren, Peng Chen, Gregor Kržmanc, Daniel Zeng, Percy S Liang, and Jure Leskovec. Prodigy: Enabling in-context learning over graphs. *Advances in Neural Information Processing Systems*, 36, 2024.

[39] Mikhail Galkin, Xinyu Yuan, Hesham Mostafa, Jian Tang, and Zhaocheng Zhu. Towards foundation models for knowledge graph reasoning. *arXiv preprint arXiv:2310.04562*, 2023. 9

[40] Wenzhuo Tang, Haitao Mao, Danial Dervovic, Saumitra Brugere, Ivan Mishra, Yuying Xie, and Jiliang Tang. Cross-domain graph data scaling: A showcase with diffusion models. 2024. URL https://arxiv.org/pdf/2406.01899. 9

[41] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 13

A Emperiment Settings.

Here we introduce the training datasets used and training hyperparameters in the experiments.

Datasets. The basic situations of the datasets we use are listed in the Table 1, 2 and 3 below.

Table 1: Graph Property Prediction Datasets Statistics.

	PCQM4MV2	OGBG-PPA	OGBG-MOLHIV	OGBG-MOLPCBA	REDDIT-MULTI-12K
Domains	Molecule	Protein	Molecule	Molecule	Social
#Graphs	3,746,619	158,100	41,127	437,929	11,929
#Total Nodes	52,970,652	38,481,540	1,048,739	11,386,154	4,669,130
#Total Edges	54,546,813	358,270,410	1,130,993	12,305,805	5,450,241
Tasks	Regression	Classification	Classification	Classification	Classification
Metrics	MSE	Accuracy	ROC-AUC	AP	Accuracy

Table 2: Node Classification Datasets Statistics.

	OGBN-ARXIV	OGBN-PRODUCTS	OGBN-PROTEINS
Domains	Citation	E-commerce	Biology
#Nodes	169,343	2,449,029	132,534
#Edges	1,166,243	61,859,140	39,561,252
Metrics	Accuracy	Accuracy	ROC-AUC

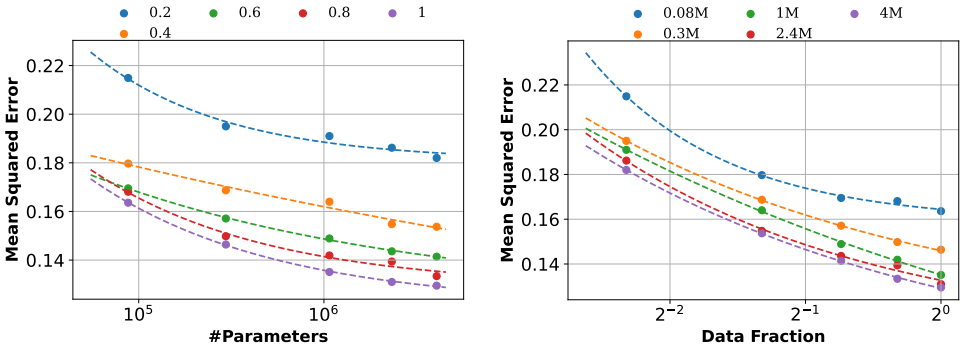
Table 3: Link Classification Datasets Statistics.

	OGBL-PPA	OGBL-COLLAB	OGBL-CITATION2
Domains	Citation	E-commerce	Biology
#Nodes	576,289	235,868	2,927,963
#Edges	30,326,273	1,285,465	30,561,187
Metrics	Hits@100	Hist@20	MRR

Training Details. For GIN and GCN models, we set the initial learning rate to 0.001 and use Adam optimizer [41] in the training process. For SAT and GraphGPS, we directly adopt the hyperparameters and optimizers that the authors present in their GitHub repos. For all the models, we opt against the learning rate with a scheduler. For OGBG-PPA, OGBN-ARXIV, OGBN-PRODUCTS, OGBN-PROTEINS, OGBL-PPA, OGBL-CITATION2 and OGBL-COLLAB datasets, the training epoch number is 50. For PCQM4MV2, OGBG-MOLHIV, OGBG-MOLPCBA, and REDDIT-MULTI-12K datasets, the training epoch number is 100. We choose the test performance with the best corresponding validation performance as the final performance of the model.

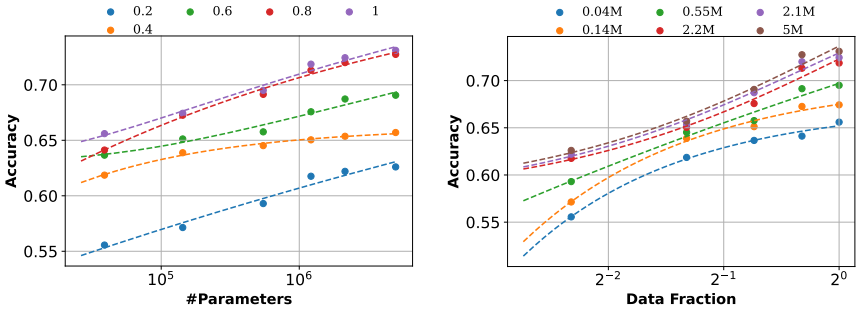
B More Results on the Validity of the Neural Scaling Law on Graphs

Here we show the scaling behaviors of **GCN** on the PCQM4MV2 and OGBG-PPA datasets. All the empirical values fit well with the neural scaling law.



(a) Model scaling on PCQM4MV2, $R^2 = 0.98$. (b) Data scaling on PCQM4MV2, $R^2 = 0.98$.

Figure 10: The model and data scaling behaviors of GCN on PCQM4MV2. For model scaling, the color corresponds to the training data fraction; for data scaling, the color corresponds to the number of model parameters. The lower mean square error is, the better model performance is. The high R^2 values indicate that the scaling law curves fit well with the empirical values.



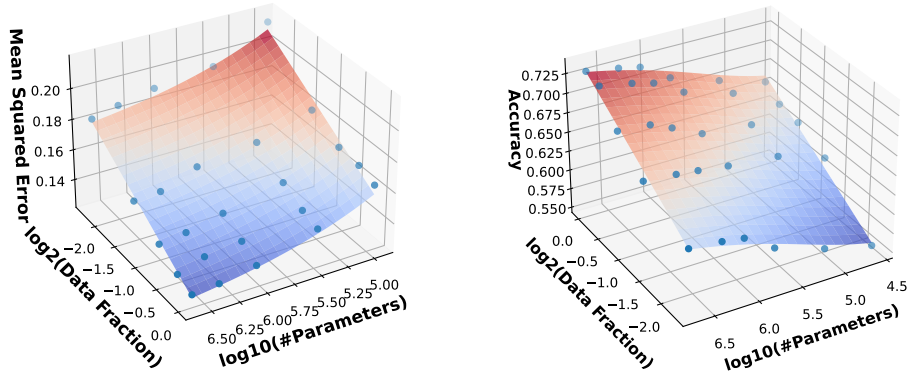
(a) Model scaling on PCQM4MV2, $R^2 = 0.98$. (b) Data scaling on PCQM4MV2, $R^2 = 0.99$.

Figure 11: The model and data scaling behaviors of GCN on OGBG-PPA. For model scaling, the color corresponds to the training data fraction; for data scaling, the color corresponds to the number of model parameters. The higher the performance is, the better the model performance is. The high R^2 values indicate that the scaling law curves fit well with the empirical values.

C Model Scaling Law on Limited Graph Data

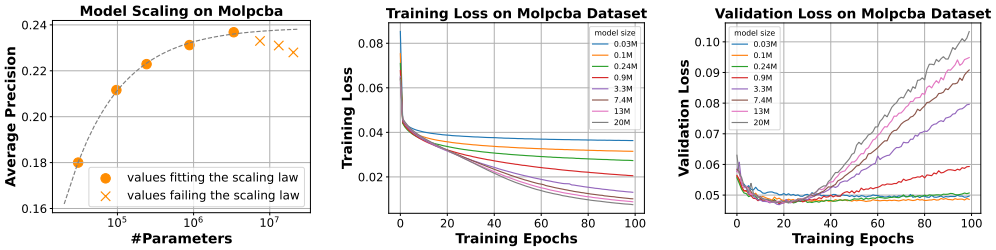
Despite the overall success, the scaling law could fail in certain situations. In particular, we find that the model scaling law could fail when the model size is too large relative to the training set size, causing the phenomenon called *model scaling collapse*. To illustrate this phenomenon, we consider an example of scaling up 5-layer GIN on OGBG-MOLPCBA dataset. Similar to the setting in Section 4, we vary the model size by changing the model width while fixing the training set size. Figure 13(a) displays the empirical accuracy of models across sizes alongside the model scaling curve. At small model sizes, the accuracy values fit with the model scaling curve well. However, when the number of parameters is around 10^7, the empirical values start dropping. Thus, the collapse of performance would make the predictions of the scaling law fail, which is referred to as model scaling collapse.

The model scaling collapse poses difficulty in building large graph models following the model scaling law. Thus, it is essential to explore why the collapse happens and how to mitigate it. In this



(a) Scaling behaviors on PCQM4MV2, $R^2 = 0.97$. (b) Scaling behaviors on OGBG-PPA, $R^2 = 0.97$.

Figure 12: The comprehensive scaling behaviors of GCN on PCQM4MV2 and OGBG-PPA datasets. For PCQM4MV2, the color corresponds to the value of mean square error; for OGBG-PPA, the color corresponds to accuracy. The surfaces predicted by the neural scaling law fit well with the empirical values on both datasets, as indicated by the high R^2 values.



(a) Model scaling on OGBG-MOLPCBA (b) Training losses of models (c) Validation losses of models

Figure 13: (a) The empirical performance of larger models falls below the prediction of model scaling law. (b)(c) The training losses and validation losses of models with varying sizes. The color corresponds to the number of model parameters. The model with larger sizes suffers more serious overfitting.

work, we conjecture that one of the potential reasons for the scaling law collapse is overfitting. To dive deeper into the problem, Figure 13(b) and 13(c) present the training loss and validation loss curves, respectively. All the training losses in Figure 13(b) keep dropping in the training process. The larger the model is, the smaller the corresponding training loss will be. Nonetheless, the validation losses of models with more than 10^7 parameters start increasing after several epochs as shown in Figure 13(c), which indicates overfitting. Moreover, the models whose performance falls behind the scaling law predictions suffer more serious overfitting. For instance, models of 7.4M, 13M, and 20M parameters overfit the most in Figure 13(c), and they are the three models on which model scaling law fails in Figure 13(a). Based on these observations, overfitting can be one potential reason for the model scaling collapse. For more results between the model size and overfitting, please refer to Appendix D.

To eliminate the effect of model scaling collapse, it is crucial to overcome the overfitting. [2] suggests that overfitting would happen if the model size is too large for the training set. To investigate whether more training data can mitigate overfitting, we train a model of fixed number parameters on different fractions of the training set and illustrate the validation loss curves in Figure 14. The model trained on 10 percent of the original OGBG-MOLPCBA dataset suffers serious overfitting as the validation loss curve suggests. However, the overfitting is significantly mitigated when the data fraction is raised to 50 percent, which indicates the effectiveness of enlarging the training set. Another evidence is that we

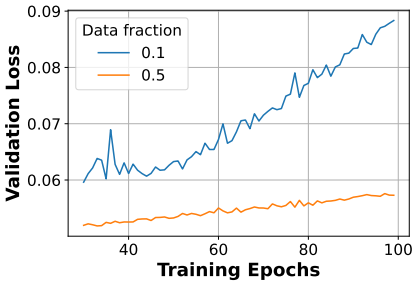


Figure 14: More training data can mitigate the overfitting problem. The model size is fixed, while the two curves correspond to different data fractions of the original OGBG-MOLPCBA training set.

have not observed overfitting or model scaling collapse in models of similar sizes (10^6 parameters) trained on larger datasets, i.e., PCQM4MV2 and OGBG-PPA in Section 4.

In graph tasks, the model scaling collapse often occurs with overfitting, which is because the graph dataset sizes are much smaller compared to those in CV or NLP. Overall, the model scaling collapse in graph tasks suggests that the current graph datasets are small for the deep graph models, leaving the models data hungry. Furthermore, it indicates that a crucial step toward the large graph model is to enlarge the size of datasets.

D More Results on the Model Scaling Collapse

Here we present more results on the relationship between the model size and the degree of overfitting. In general, the observation in the below figures indicate that the lager model is, the more serious it would overfit.

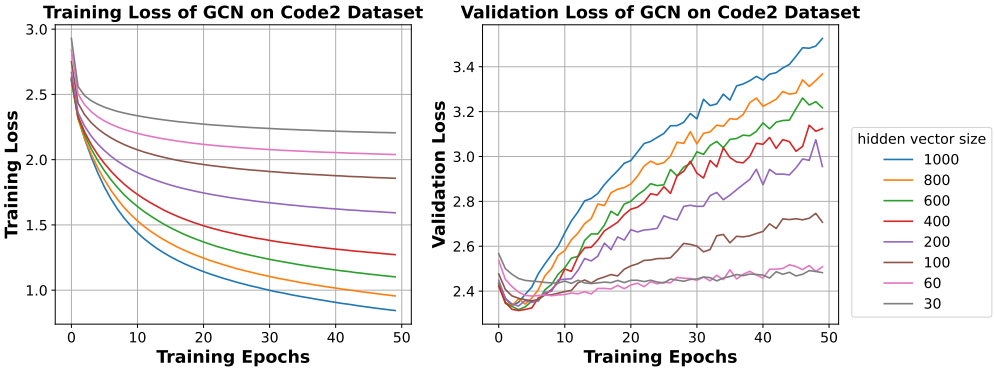


Figure 15: The training losses and validation losses of GCN with varying sizes on Code2 Dataset. The color corresponds to the number of model parameters. The figure shows that the model with larges size has more serious overfitting.

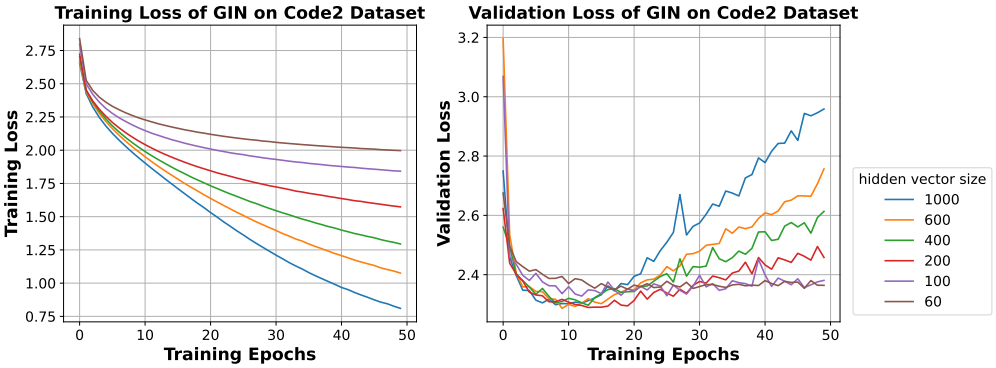


Figure 16: The training losses and validation losses of GIN with varying sizes on Code2 Dataset. The color corresponds to the number of model parameters. The figure shows that the model with larges size has more serious overfitting.

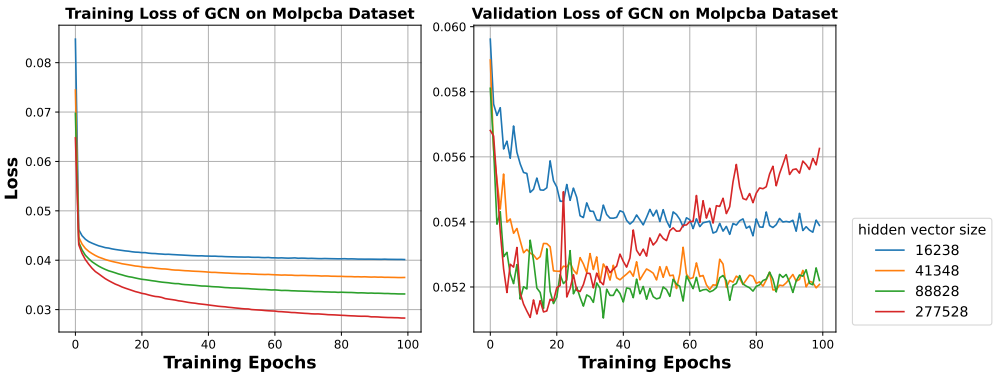
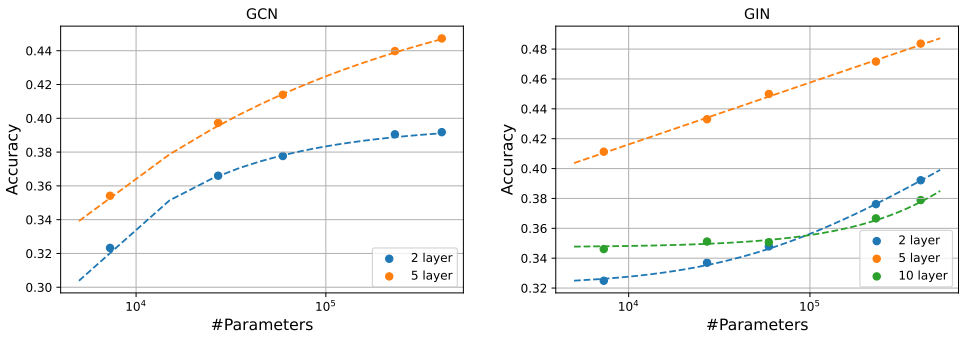


Figure 17: The training losses and validation losses of GCN with varying sizes on OGBG-PPA Dataset. The color corresponds to the number of model parameters. The figure shows that the model with larges size has more serious overfitting.

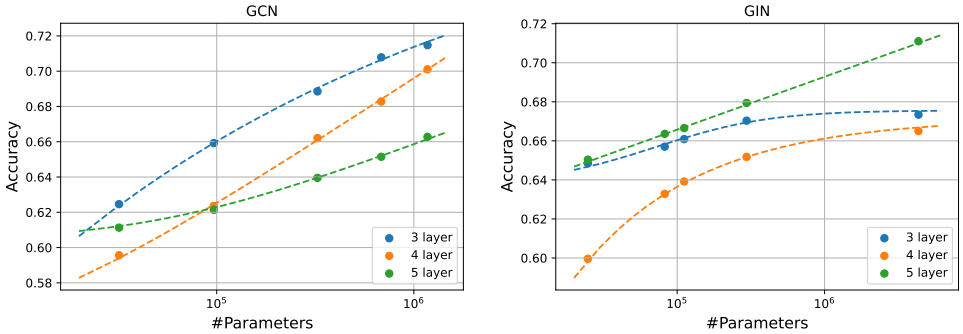
E More Results on the Impacts of Model Depths

We present the experiment results of model scaling behaviors with varying model depths on more datasets. In the following figures, models with different depths exhibit distinct model scaling behaviors.



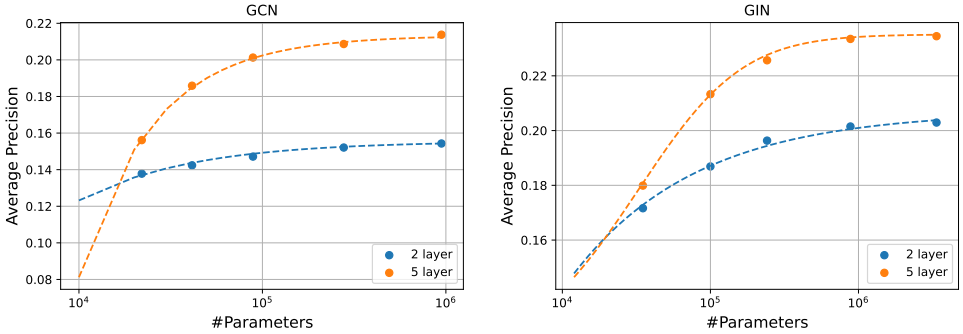
(a) Model scaling of GCN with different layers. (b) Model scaling of GIN with different layers.

Figure 18: Model scaling behaviors of GNNs with varying model depths on REDDIT-MULTI-12K. The color corresponds to the model layer. Both GIN and GCN exhibit distinct scaling behaviors with different model depths.



(a) Model scaling of GCN with different layers. (b) Model scaling of GIN with different layers.

Figure 19: Model scaling behaviors of GNNs with varying model depths on OGBG-MOLHIV. The color corresponds to the model layer. Both GIN and GCN exhibit distinct scaling behaviors with different model depths.

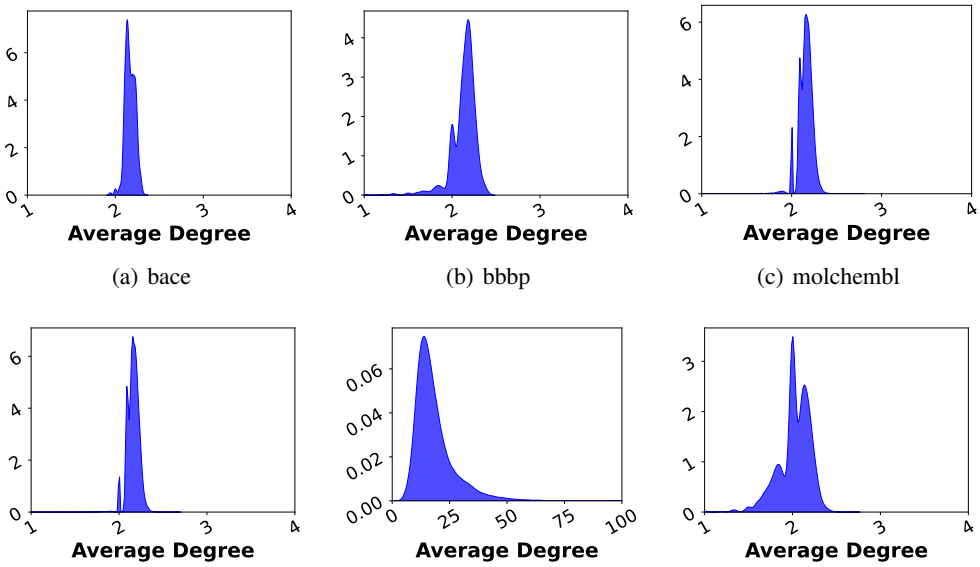


(a) Model scaling of GCN with different layers. (b) Model scaling of GIN with different layers.

Figure 20: Model scaling behaviors of GNNs with varying model depths on OGBG-MOLPCBA. The color corresponds to the model layer. Both GIN and GCN exhibit distinct scaling behaviors with different model depths.

F The Average Degree Distributions of More Datasets

Here we present the average degree distributions of more datasets in Figure 21. The samples in one dataset tend to have similar average degrees, further supporting our argument in Section 6.



(a) bace (b) bbbp (c) molchembl

(d) molpcba (e) Ogbg-ppa (f) tox21

Figure 21: The average degree distribution of more datasets.

G Results with the Number of Nodes as metric for Section 6

Here we present more results of the comparison of the data metric of the data scaling law. Figure 22 shows that the number of nodes as metric could also give better fitting than the number of graphs, supporting the conclusion of Experiment 1 in Section 6. Figure 23 shows that training subsets with the same number of total nodes but different numbers of graphs tend to have similar scaling behaviors, supporting the conclusion of Experiment 2 in Section 6.

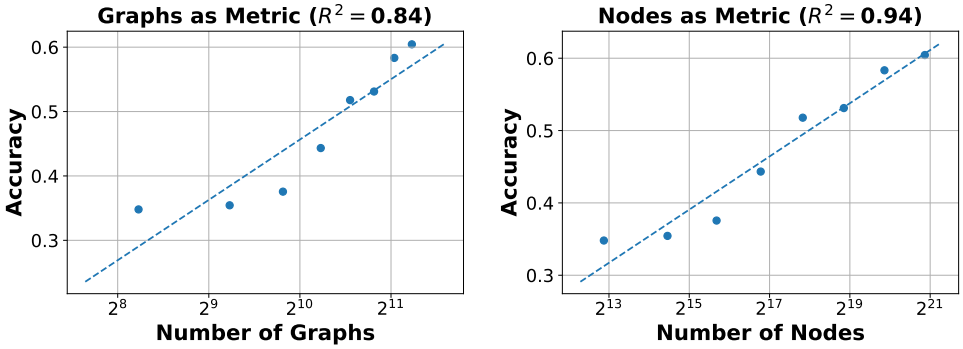


Figure 22: Comparison between the number of graphs and the number of nodes as the data metric

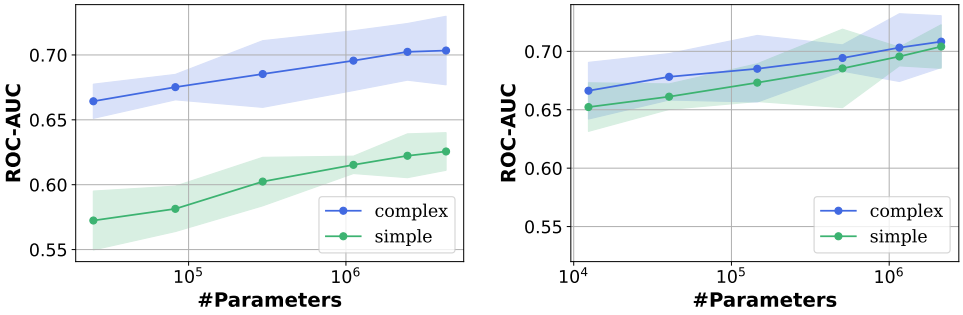


Figure 23: Model scaling curves of GIN on two training subsets of different sizes. The shade corresponds to the performance variance of repeated experiments.

H The Data Scaling Law of Node Classification and Link Prediction with the Number of Nodes as Metric

Here we present the results of the data scaling law on node classification and link prediction tasks with the number of nodes as the data metric. From Figure 24, we find that the number of nodes also gives good fitting results as the data metric.

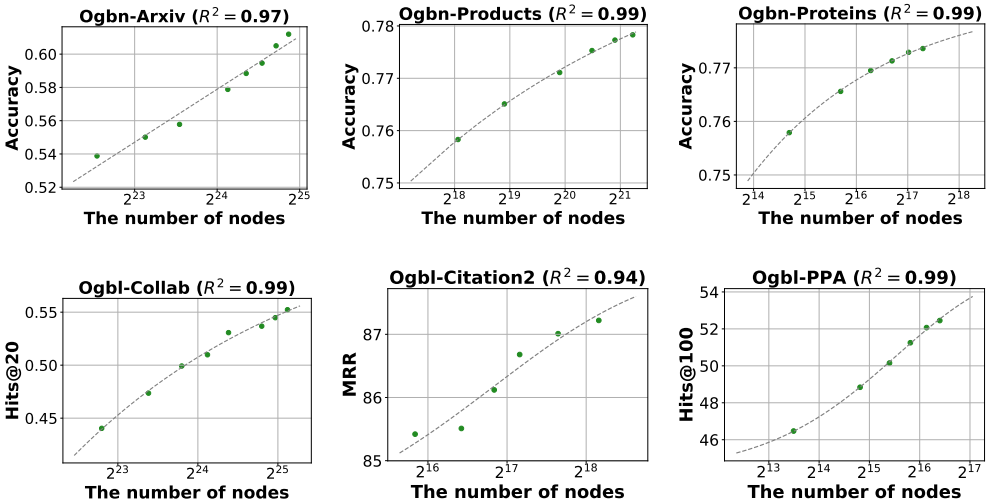


Figure 24: Both node classification and link prediction follow the data scaling law with the total node number as the data metric.

I How do Our Findings Connect to Compute Scaling and Chinchilla Scaling Law

Here we first show that the training compute can be calculated when we know the total edge number and the model architecture. We consider the general Message Passing Neural Networks, which can be formulated as above:

$$m_v^{t+1} = U(h_v^t, \sum_{w \in N(v)} M(h_v^t, h_w^t, e_{vw})) \tag{6}$$

where U and M can be assumed to contain x and y add-multiply operations, respectively. Suppose the total node number in the dataset is N, the total edge number is E, and the model has L layers. For node i, aggregating messages from its neighbors needs $x + yd_i$ add-multiply operations, where d_i is the degree of node i. For the whole datasets, there would be $\sum_i x + yd_i = 2(1 + 1/d_a)yE$, where d_a is the average degree of the dataset. For the whole model, one feed-forward pass needs $2(1 + 1/d_a)yEL$ add-multiply operations. Thus, we have shown that we can calculate the training FLOPs if we know the total edge number in the dataset.

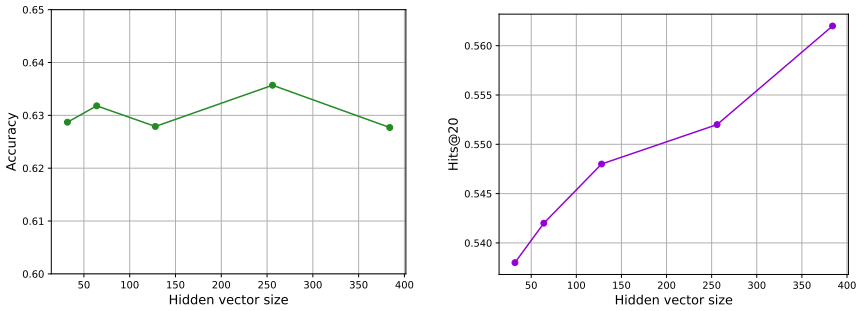
Chinchilla scaling law aims to allocate the best training set size and the model size when the compute cost is fixed [15]. The graph scaling law developed in our paper can be easily extended to calculate optimal parameter numbers or dataset size as shown below. For simplicity, we just consider Equation 4 here. But the results can be easily extended to Equation 5. For Equation 4 we have

$$\epsilon = \epsilon_\infty + A/N^\alpha + B/D^\beta$$

where N is the model size, and D is the training set size. From the discussion above and in [2], we know the FLOPs for GIN are about $6\frac{1+d_a}{d_a}EN$, where E is the number of edges and d_a is the average degree of the training set. For fixing the compute budget $C = 6\frac{1+d_a}{d_a}EN$, we can get $N_{opt} = G(\frac{d_a C}{6(1+d_a)})^a$, $E_{opt} = G^{-1}(\frac{d_a C}{6(1+d_a)})^b$ following [2], where $G = (\frac{\alpha A}{\beta B})^{\frac{1}{\alpha+\beta}}$, $a = \frac{\beta}{\alpha+\beta}$, $b = \frac{\alpha}{\alpha+\beta}$. We can find that there is a power-law relationship between the optimal parameters/ training set size and the FLOPs.

J The Model Scaling Behaviors of Node Classification and Link Prediction

Here we investigate the model scaling behaviors of node classification and link prediction tasks. The observations in the below figure show that SEAL trained on OGBL-COLLAB has obvious performance gains when the model size is increased, while this is not the case for GraphSAGE on OGBN-ARXIV. We leave further exploration of these topics for future works.



(a) Model scaling behaviors on OGBN-ARXIV. (b) Model scaling on OGBL-COLLAB.

Figure 25: In (b), the model performance has obvious gains on the link prediction task when the hidden vector size is increased, while it is not for the node classification task in (a).

K Broader Impacts

In this paper, we investigate neural scaling laws on graphs, which describe how the model performance will grow with increasing model and data sizes. Neural scaling laws can provide instructions for

building large graph models, predicting the costs of computing and data labeling to achieve the expected performance. Moreover, it may incentivize more investments in data collecting and even the construction of larger graph datasets for various tasks. We anticipate that our findings will serve as essential guidance for the development of large graph models.