Evaluating Self-Supervised Learned Molecular Graphs

Anonymous Authors[1]

Abstract

Because of data scarcity in real-world scenarios, obtaining pre-trained representations via self-supervised learning (SSL) has attracted increasing interest. Although various methods have been proposed, it is still under-explored what knowledge the networks learn from the pre-training tasks and how it relates to downstream properties. In this work, with an emphasis on chemical molecular graphs, we fill in this gap by devising a range of node-level, pair-level, and graph-level probe tasks to analyse the representations from pre-trained graph neural networks (GNNs). We empirically show that: 1. Pre-trained models have better downstream performance compared to randomly-initialised models due to their improved the capability of capturing global topology and recognising substructures. 2. However, randomly initialised models outperform pre-trained models in terms of retaining local topology. Such information gradually disappears from the early layers to the last layers for pre-trained models.

1. Introduction

Self-Supervised Learning (SSL) pre-training has opened up the opportunity to effectively utilise vast amount of unlabelled data to improve downstream tasks where labels are limited. In natural language processing, language models like GPT-3 (Brown et al., 2020), Megatron (Shoeybi et al., 2019), and Gopher (Rae et al., 2021) can automatically rediscover the classical NLP pipeline in an interpretable and localisable way (Tenney et al., 2019). They can also achieve substantial improvements in a wide range of NLP tasks. In computer vision, self-supervised learning approaches such as contrastive learning (Chen et al., 2020b; He et al., 2020), bootstrapping (Grill et al., 2020) and masking (He et al., 2022) are shown to obtain competitive performance on widely-used benchmarks like ImageNet. DINO (Caron et al., 2021a) shows that a self-supervised vision transformer (ViT) automatically learns class-specific features for unsupervised object segmentation.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

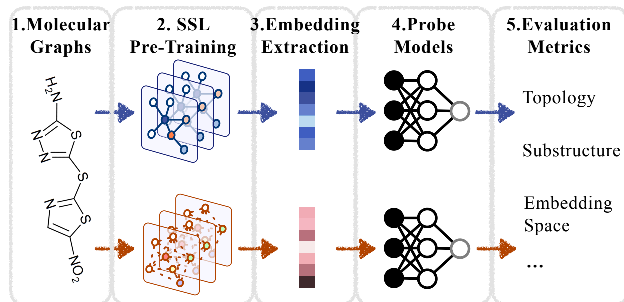

Figure 1. Overview of *GraphEval*. Given molecular graphs, we train GNNs to predict SSL proxy objectives. We then extract embeddings of (possibly unseen) graphs using pre-trained models, which form the inputs for probe models, trained and evaluated on the designed metrics.

Motivated by the successful applications of self-supervised learning, pre-training GNNs on unlabelled structured data has attracted increasing interest (Liu et al., 2021a; Xie et al., 2021). However, it is still under-explored what knowledge the networks learn during the pre-training and how it relates to downstream properties. In this work, with an emphasis on chemical molecules, we fill in this gap by devising: (1) a range of {node-, pair-, graph-} level metrics; (2) substructure detection; (3) embedding space characterisation, to analyse the representations from pre-trained GNNs. Our main insights are summarised as follows:

- Pre-trained representations are better at capturing global topological structure while losing the local information;
- Pre-trained models can well recognise molecular substructures that are correlated with properties;

2. Preliminaries and Settings

We first introduce the basics of graphs and GNNs, then elaborate on the pre-training and probes.

Graph. A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consists of a set of nodes $\mathcal{V}$ and edges $\mathcal{E}$. In molecular graphs, nodes are atoms and edges are bonds. We use $\boldsymbol{x}_u$ and $\boldsymbol{x}_{uv}$ to denote the feature of node u and of the bond feature between nodes $[u, v]$, respectively. For notation simplicity, we use an adjacency matrix

Table 1. Performance on molecular property predictions using probes, with (w/) or without (w/o) fine-tuning (FT). For each set of random/pre-trained embeddings, we report the ROC-AUC scores over 8 datasets consisting of 678 binary tasks, where the score of each task is averaged over three independent runs. We **bold** the best and underline the worst performance of each dataset.

FT	Random	AttrMask	GPT-GNN	InfoGraph	ContextPred	G-Contextual	G-Motif	GraphCL	JOAO	JOAOv2
w/o	58.85	62.18	61.43	61.94	59.58	**64.63**	62.59	63.00	60.99	62.31
w/	67.21	70.16	68.27	70.10	**70.89**	69.21	70.14	70.64	69.57	70.21

$\mathbf{A} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ to represent the graph, where $\mathbf{A}[u, v] \neq 0$ if the nodes (u, v) are connected.

GNN. There has been emerging research interest in exploring molecular graph representations (Corso et al., 2020; Duvenaud et al., 2015; Gilmer et al., 2017; Liu et al., 2018; Yang et al., 2019). Graph neural networks are widely-adopted for encoding molecular graphs. A prototypical GNN uses messaging passing (Gilmer et al., 2017), where it updates atom-level representations based on their neighbourhoods. More specifically, let $\boldsymbol{h}_u^0 = \boldsymbol{x}_u$ be the input atom feature, we have:

$$\boldsymbol{m}_u^{t+1} = \sum_{v:\mathbf{A}[u,v]\neq 0} M_t(\boldsymbol{h}_u^t, \boldsymbol{h}_v^t, \boldsymbol{x}_{uv}), \quad \boldsymbol{h}_u^{t+1} = U_t(\boldsymbol{h}_u^t, \boldsymbol{m}_u^{t+1}) \tag{1}$$

where M_t and U_t are the message functions and vertex update functions, respectively. By repeating message passing for T steps, we can encode the information of the T-hop neighbourhood for each atom. We use a readout function R to pool node-level representations for graph-level prediction: $\hat{y} = R(\{\boldsymbol{h}_u^T | u \in \mathcal{V}\})$. In this work, we follow the research line of SSL on molecular graphs (Hu et al., 2020a; Liu et al., 2022; You et al., 2020) and adopt the Graph Isomorphism Network (GIN) (Xu et al., 2019) as the backbone model (modified in (Hu et al., 2020a) as to incorporate edge features during message passing).

Pre-Training. We use ten methods for Graph SSL, including EdgePred (Hamilton et al., 2017), InfoGraph (Sun et al., 2020), GPT-GNN (Hu et al., 2020b), AttrMask (Hu et al., 2020a), ContextPred (Hu et al., 2020a), G-{Contextual, Motif} (Rong et al., 2020), GraphCL (You et al., 2020), JOAO-{·,v2} (You et al., 2021) for pre-training. We follow the experimental settings and pre-training recipes reported in the original literature. For a fair comparison, we pre-train the same GIN model on the same data splits. Specifically, we randomly select 50k qualified molecules from the GEOM dataset (Axelrod & Gomez-Bombarelli, 2020). Once the pre-training finished, we extract the embeddings based on the saved weights and pass them to the probe tasks.

Probe. We use probe models (Liu et al., 2019) to study whether self-supervised learned representations encode helpful structural information about graphs. Concretely, we use a graph neural network to extract graph representations and train a shallow model to make predictions with these fixed node and graph embeddings. A common choice of the probe model (Hewitt & Liang, 2019) is either a linear projection or a multi-layer perceptron (MLP). We choose an MLP with one hidden layer to enable capturing the non-linear relations. We set the hidden size to 300 and apply the ReLU activation. We use scaffold splitting to split data into 80%/10%/10% for the training/validation/testing set. The training procedure runs for 100 epochs with a learning rate of $1e^{-3}$. We select the best model based on the validation set. All the results are averaged across three independent runs.

As follows, we show the effectiveness of SSL methods in downstream tasks and systematically study the knowledge that the networks learn from the pre-training tasks:

- In Sec. 3, we evaluate SSL learned embeddings on molecular biochemical property, demonstrating that such substantial improvements with linear models and fine-tuning are not much relevant.
- In Sec. 4, we probe a wide range of structural and topological metrics based on the embeddings. We find that pre-trained embeddings are better at capturing global topological property, and randomised variants surprisingly outperform restoring local geometry.
- In Sec. 5, we demonstrate that pre-trained embeddings are better at predicting the counts of molecular substructures, *e.g.*allylic and benzene. We hypothesise that the superior performance of pre-trained embeddings for molecular biochemical property prediction comes from the fact that SSL pre-training help better capture the substructure existence (Alsentzer et al., 2020; Bouritsas et al., 2020).

3. Biochemical Property Measure

We first use probe models to evaluate pre-trained embeddings on predicting molecular biochemical properties. Following previous graph SSL work (Hu et al., 2020a; You et al., 2020), we validate the quality of these embeddings on eight molecular datasets consisting of 678 binary property prediction tasks (Hu et al., 2021; Wu et al., 2018). As previously described in Sec. 2, for the setting of without fine-tuning (“w/o FT”), we update the probe models with fixed embeddings; with fine-tuning (“w FT”), both the pre-trained GNNs and the randomised probe models will be updated. We report the results in Table 1.

Table 2. Performance on the topological metrics predictions. We report the mean square or the cross entropy loss (*i.e.*, the smaller the better), over all 8 downstream datasets. We **bold** the best and underline the worst performance of each metric. We have summarised the percentage where SSL pre-trained embeddings fail to outperform the random embeddings.

Metrics	Node			Pair			Graph			
Pre-training	Degree	Centrality	Clustering	Link	Jaccard	Katz	Diameter	Connectivity	Cycle	Assortativity
–	**0.001**	1.199	0.297	**31.05**	1.879	<u>2.828</u>	<u>222.6</u>	<u>0.226</u>	<u>6.351</u>	<u>0.158</u>
AttrMask	0.015	1.307	0.424	32.23	2.029	2.634	164.7	**0.178**	6.075	**0.102**
GPT-GNN	3.032	1.380	0.505	41.44	<u>2.541</u>	2.374	178.8	0.247	9.222	0.166
InfoGraph	1.298	1.242	**0.296**	41.15	2.273	2.238	**83.24**	0.204	6.169	0.159
ContextPred	<u>5.498</u>	<u>1.626</u>	0.316	37.78	2.286	2.413	183.0	0.194	8.691	0.108
G-Motif	3.085	1.372	<u>0.531</u>	<u>51.83</u>	2.363	2.758	98.21	0.268	7.333	0.182
G-Contextual	0.036	1.242	0.403	33.55	**1.773**	2.660	113.6	0.170	**5.330**	0.045
GraphCL	0.854	**1.110**	0.461	34.97	1.863	**2.271**	89.79	0.226	6.191	0.152
JOAO	0.637	1.268	0.412	33.67	2.084	2.307	89.38	0.214	5.960	0.142
JOAOv2	0.591	1.272	0.463	32.81	2.054	2.340	88.27	0.217	5.964	0.148
SSL Worse	100%	89%	89%	100%	78%	0%	0%	0%	0%	0%

Results and Findings. As shown in Table 1, most of SSL pre-trained embeddings outperform the randomised peers both under fixed and non-fixed settings. Compared with fixed embeddings, tuning the pre-trained model weights will bring more substantial performance gains due to introducing more flexibility. However, in general, better performance at fixed embeddings does not accompany higher fine-tuning scores. For instance, embeddings pre-trained with "ContextPred" have the second-lowest score with fixed scenarios while perform the best after end-to-end fine-tuning. The correlation between the two sets of score rankings is 0.25, which questions the conventional approach's rationale for evaluating the quality of learned embedding with linear models (He et al., 2022).

4. Topological Property Measure

We evaluate the pre-trained embeddings on metrics emphasising topological properties at multiple scales, which are based on the {node-, pair-, and graph-} level statistics. Many of these metrics are used as features in traditional machine learning pipelines on graphs prior to the advent of deep learning (Hamilton, 2020). We first provide descriptions of these metrics, then present results and findings.

Results and Findings. We report the results in Table 2. We observe that the randomised embeddings retain the local structural information well and outperform all the pre-trained embeddings. On the other hand, the pre-trained embeddings perform well when performing metrics related to the graph's global topology. For pair-level statistics, randomised embeddings perform better when the metric itself is more about local structure, *e.g.*link prediction, and vice versa. We do not observe that there exists a dominant pre-training method that perform universally well w.r.t. other methods. There are some connections between the pre-training tasks and the performance on different metrics:

- Contextual proxy (i.e., G-Contextual) is particularly helpful for Jaccard coefficient prediction because of the similarity of the pre-training objective and metric measure (neighbourhood overlap);
- Complicated design of augmentations (used in contrastive-based SSL, *i.e.*JOAO) do not bring substantial improvements in storing graph-level topological information.

5. Substructure Awareness Measure

Certain *substructures* usually reflect some properties at node and graph levels (Girvan & Newman, 2002). For instance, molecules containing benzene rings usually have similar physical (*e.g.*solvent) and chemical (*e.g.*aromaticity) properties (McMurry, 2014). On this basis, prediction (Alsentzer et al., 2020) and modelling (Bouritsas et al., 2020) of substructures have been proven effective for improving model expressiveness and downstream performance.

Molecular substructure. Instead of defining in an implicit or handcrafted manner, as in previous studies, a natural definition of substructure in molecules is the substituent or moiety that performs certain functions in chemical/biological reactions. Here we investigate 24 substructures which can be divided into three groups:

- **Rings**: Benzene, Beta lactams, Epoxdie, Furan, Imidazole, Morpholine, Oxazole, Piperdine, Piperdine, Pyri-

Table 3. Cramér's V between molecular substructure counts and biochemical properties, averaged over 678 property prediction tasks (*i.e.*, "Avg(Task)") or eight datasets (*i.e.*, "Avg(Data)"). We also calculate the Pearson rank correlation (ρ) between the performance on recognising the substructure and predicting properties.

Name	Type	Avg (Task)	Avg (Data)	ρ
allylic	Site	0.1144	0.1024	0.709
benzene	Ring	0.1630	0.1227	0.576
amide	Group	0.0881	0.1336	0.468
ether	Group	0.1034	0.1083	0.552
halogen	Group	0.1721	0.1086	0.515

dine, Tetrazole, Thiazole, Thiophene

- **Functional Groups**: Amides, Amidine, Azo, Ether Guanidine, Halogens, Hydroxylamine, Imide, Oxygens (including phenoxy), Urea
- **Redox Active Sites**: Allylic (excluding steroid dienone)

Each substructure might have unique effect on the downstream properties. For instance, forming with a simple cycle of atoms and bonds, a ring might lock particular atoms with distinct 3D structure therefore some of its stereochemistry properties such as chirality are determined, and chirality-aware modelling is proven beneficent in predicting molecular properties (Adams et al., 2022). We first apply "Cramér's V" to measure how significant the substructures affect the molecular properties.

Cramér's V quantifies the strength of the association between the molecular substructure counts (*i.e.*, chemical fragments) and their biochemical properties. It is defined as:

$$V = \sqrt{\chi^2 / \left(n \cdot \min(k-1, r-1)\right)} = \sqrt{\chi^2/n} \quad (r \equiv 2) \tag{2}$$

where n is the sample size, k and r are the total number of substructure counts and property categories (binary), respectively. The Chi-squared statistics χ^2 is then calculated as:

$$\chi^2 = \sum_{i,j} \left(n_{(i,j)} - n_{(i,\cdot)} \cdot n_{(\cdot,j)}/n\right)^2 \Big/ \left(n_{(i,\cdot)} \cdot n_{(\cdot,j)}/n\right) \tag{3}$$

where $n_{(i,j)}$ is the total occurrence for the pair of (i, j). Here i is the specific count of a certain substructure, and j represents the certain outcome of a molecular biochemical property. Cramér's V value ranges from 0 to 1, representing the associated strength between two categorical variables.

Results and Findings. We calculate the Cramér's V, and report the five substructures that are mostly correlated with downstream properties in Table 3. We observe that certain molecular substructures are good indicators of their

Table 4. Performance on substructure detection. We **bold** the best and underline the worst performance of each substructure. It is clear to see that contrastive based method (GraphCL, JOAOv2) perform quite well in recognising these substructures.

Pre-training	allylic	amide	benzene	ether	halogen
–	3.516	18.948	3.964	6.071	3.652
AttrMask	3.371	12.932	2.860	4.958	1.192
GPT-GNN	2.808	15.736	2.938	5.932	2.912
InfoGraph	2.577	5.535	1.959	3.657	2.819
ContextPred	4.386	18.251	3.583	7.045	2.908
G-Motif	2.452	4.015	2.116	3.507	1.125
G-Contextual	2.196	5.938	1.926	**2.900**	0.759
GraphCL	**2.088**	3.922	**1.722**	3.766	0.798
JOAO	2.385	4.030	1.746	3.376	**0.694**
JOAOv2	2.122	**3.865**	1.773	3.388	0.695
SSL Worse	11%	0	0	11%	0

biochemical properties. Based on such facts, we train the probe models to predict the counts of substructures for all the molecules from the eight datasets. We report the test scores in Table 4. As noticed, all the pre-trained embeddings outperform random variants in terms of detecting the existence of substructures.

We also calculate the Pearson rank correlation ρ between the performance on downstream tasks and the performance on substructure detection of the SSL pre-trained embeddings. A strong positive correlation indicate that embeddings that are with better capability of detecting these substructures . Based on the observations of (1) molecular substructures are highly related with downstream biochemical properties; (2) embeddings that perform better in property predictions are usually with better substructure awareness; we conjecture that the performance gains from SSL pre-training might be from their capabilities of identifying graph substructures.

We find that: 1) substructure counts is highly correlated with the molecular properties; 2) the pre-trained embeddings are good at counting the substructures and predicting the properties. Consequently, we would like to measure that how well we can infer the properties solely based on the substructure counts (in Appendix).

6. Discussion

In this work, we conduct a collection of probe tasks and analysis on evaluating the self-supervised learned graph embeddings. We conclude the performance gains introduced by the SSL pre-training come from a better awareness of global topology and substructures. The pre-trained message passing weights, help capture the hierarchical while hurdle the local information. A better design on the message passing module remains an open problem.

References

Adams, K., Pattanaik, L., and Coley, C. W. Learning 3d representations of molecular chirality with invariance to bond rotations. In *ICLR*, 2022.

Alain, G. and Bengio, Y. Understanding intermediate layers using linear classifier probes. In *ICLR Workshop*, 2017.

Alsentzer, E., Finlayson, S., Li, M., and Zitnik, M. Subgraph neural networks. In *NeurIPS*, 2020.

Axelrod, S. and Gomez-Bombarelli, R. Geom: Energy-annotated molecular conformations for property prediction and molecular generation. *arXiv:2006.05531*, 2020.

Bouritsas, G., Frasca, F., Zafeiriou, S., and Bronstein, M. M. Improving graph neural network expressivity via subgraph isomorphism counting. *arXiv:2006.09252*, 2020.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. In *NeurIPS*, 2020.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging properties in self-supervised vision transformers. *arXiv:2104.14294*, 2021a.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021b.

Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., and Sutskever, I. Generative pretraining from pixels. In *ICML*, 2020a.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *ICML*, 2020b.

Chen, X., Xie, S., and He, K. An empirical study of training self-supervised vision transformers. *arXiv:2104.02057*, 2021.

Conneau, A. and Kiela, D. Senteval: An evaluation toolkit for universal sentence representations. In *LREC*, 2018.

Corso, G., Cavalleri, L., Beaini, D., Liò, P., and Veličković, P. Principal neighbourhood aggregation for graph nets. In *NeurIPS*, 2020.

Dohan, D., Gane, A., Bileschi, M. L., Belanger, D., and Colwell, L. Improving protein function annotation via unsupervised pre-training: Robustness, efficiency, and insights. In *KDD*, 2021.

Duvenaud, D., Maclaurin, D., Aguilera-Iparraguirre, J., Gómez-Bombarelli, R., Hirzel, T., Aspuru-Guzik, A., and Adams, R. P. Convolutional networks on graphs for learning molecular fingerprints. In *NeurIPS*, 2015.

Elnaggar, A., Heinzinger, M., Dallago, C., Rihawi, G., Wang, Y., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M., et al. Prottrans: towards cracking the language of life's code through self-supervised deep learning and high performance computing. *IEEE PAMI*, 2021.

Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., and Dahl, G. E. Neural message passing for quantum chemistry. In *ICML*, 2017.

Girvan, M. and Newman, M. Community structure in social and biological networks. *PNAS*, 2002.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., Doersch, C., Pires, B. A., Guo, Z. D., Azar, M. G., et al. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020.

Hamilton, W. L. Graph representation learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 14(3):1–159, 2020.

Hamilton, W. L., Ying, R., and Leskovec, J. Inductive representation learning on large graphs. In *NeurIPS*, 2017.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. B. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. B. Masked autoencoders are scalable vision learners. In *CVPR*, 2022.

Hendricks, L. A., Mellor, J., Schneider, R., Alayrac, J., and Nematzadeh, A. Decoupling the role of data, attention, and losses in multimodal transformers. *TACL*, 2021.

Hewitt, J. and Liang, P. Designing and interpreting probes with control tasks. In *EMNLP*, 2019.

Hewitt, J. and Manning, C. D. A structural probe for finding syntax in word representations. In *NAACL*, 2019.

Hu, W., Liu, B., Gomes, J., Zitnik, M., Liang, P., Pande, V., and Leskovec, J. Strategies for pre-training graph neural networks. In *ICLR*, 2020a.

Hu, W., Fey, M., Zitnik, M., Dong, Y., Ren, H., Liu, B., Catasta, M., and Leskovec, J. Open graph benchmark: Datasets for machine learning on graphs. In *NeurIPS*, 2021.

Hu, Z., Dong, Y., Wang, K., Chang, K.-W., and Sun, Y. Gpt-gnn: Generative pre-training of graph neural networks. In *KDD*, 2020b.

Jawahar, G., Sagot, B., and Seddah, D. What does BERT learn about the structure of language? In *ACL*, 2019.

Jiang, D., Wu, Z., Hsieh, C.-Y., Chen, G., Liao, B., Wang, Z., Shen, C., Cao, D., Wu, J., and Hou, T. Could graph neural networks learn better molecular representation for drug discovery? a comparison study of descriptor-based and graph-based models. *Journal of cheminformatics*, 2021.

Kassner, N. and Schütze, H. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *ACL*, 2020.

Li, C., Yang, J., Zhang, P., Gao, M., Xiao, B., Dai, X., Yuan, L., and Gao, J. Efficient self-supervised vision transformers for representation learning. *arXiv preprint arXiv:2106.09785*, 2021.

Liu, N. F., Gardner, M., Belinkov, Y., Peters, M. E., and Smith, N. A. Linguistic knowledge and transferability of contextual representations. In *NAACL*, 2019.

Liu, S., Demirel, M. F., and Liang, Y. N-gram graph: Simple unsupervised representation for graphs, with applications to molecules. In *NeurIPS*, 2018.

Liu, S., Wang, H., Liu, W., Lasenby, J., Guo, H., and Tang, J. Pre-training molecular graph representation with 3d geometry. In *ICLR*, 2022.

Liu, X., Zhang, F., Hou, Z., Mian, L., Wang, Z., Zhang, J., and Tang, J. Self-supervised learning: Generative or contrastive. *IEEE TKEE*, 2021a.

Liu, Y., Pan, S., Jin, M., Zhou, C., Xia, F., and Yu, P. S. Graph self-supervised learning: A survey. *arXiv:2103.00111*, 2021b.

Lü, L. and Zhou, T. Link prediction in complex networks: A survey. *Physica A: statistical mechanics and its applications*, 2011.

McMurry, J. E. *Organic chemistry with biological applications*. Cengage Learning, 2014.

Rae, J. W., Borgeaud, S., Cai, T., Millican, K., Hoffmann, J., Song, F., Aslanides, J., Henderson, S., Ring, R., Young, S., et al. Scaling language models: Methods, analysis & insights from training gopher. *arXiv:2112.11446*, 2021.

Rao, R., Bhattacharya, N., Thomas, N., Duan, Y., Chen, X., Canny, J., Abbeel, P., and Song, Y. S. Evaluating protein transfer learning with tape. In *NeurIPS*, 2019.

Resnick, C., Zhan, Z., and Bruna, J. Probing the state of the art: A critical look at visual representation evaluation. *arXiv:1912.00215*, 2019.

Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., Guo, D., Ott, M., Zitnick, C. L., Ma, J., et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *PNAS*, 2021.

Rong, Y., Bian, Y., Xu, T., Xie, W., Wei, Y., Huang, W., and Huang, J. Self-supervised graph transformer on large-scale molecular data. In *NeurIPS*, 2020.

Shoeybi, M., Patwary, M., Puri, R., LeGresley, P., Casper, J., and Catanzaro, B. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv:1909.08053*, 2019.

Stärk, H., Beaini, D., Corso, G., Tossou, P., Dallago, C., Günnemann, S., and Liò, P. 3d infomax improves gnns for molecular property prediction. *arXiv:2110.04126*, 2021.

Sun, F.-Y., Hoffmann, J., Verma, V., and Tang, J. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *ICLR*, 2020.

Tenney, I., Das, D., and Pavlick, E. BERT rediscovers the classical NLP pipeline. In *ACL*, 2019.

Villegas-Morcillo, A., Makrodimitris, S., van Ham, R. C., Gomez, A. M., Sanchez, V., and Reinders, M. J. Unsupervised protein embeddings outperform hand-crafted sequence and structure features at predicting molecular function. *Bioinformatics*, 2021.

Wang, A., Hula, J., Xia, P., Pappagari, R., McCoy, R. T., Patel, R., Kim, N., Tenney, I., Huang, Y., Yu, K., Jin, S., Chen, B., Durme, B. V., Grave, E., Pavlick, E., and Bowman, S. R. Can you tell me how to get past sesame street? sentence-level pretraining beyond language modeling. In *ACL*, 2019.

Wang, H., Liu, Q., Yue, X., Lasenby, J., and Kusner, M. J. Unsupervised point cloud pre-training via occlusion completion. In *ICCV*, 2021.

Wang, K., Zhang, Y., Yang, D., Song, L., and Qin, T. Gnn is a counter? revisiting gnn for question answering. In *ICLR*, 2022.

Watts, D. J. and Strogatz, S. H. Collective dynamics of 'small-world' networks. *Nature*, 1998.

Wu, L., Lin, H., Gao, Z., Tan, C., Li, S., et al. Self-supervised on graphs: Contrastive, generative, or predictive. *arXiv:2105.07342*, 2021.

Wu, Z., Ramsundar, B., Feinberg, E. N., Gomes, J., Geniesse, C., Pappu, A. S., Leswing, K., and Pande, V. Moleculenet: a benchmark for molecular machine learning. *Chemical Science*, 2018.

Xie, Y., Xu, Z., Zhang, J., Wang, Z., and Ji, S. Self-supervised learning of graph neural networks: A unified review. *arXiv:2102.10757*, 2021.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. How powerful are graph neural networks? In *ICLR*, 2019.

Yang, K., Swanson, K., Jin, W., Coley, C., Eiden, P., Gao, H., Guzman-Perez, A., Hopper, T., Kelley, B., Mathea, M., et al. Analyzing learned molecular representations for property prediction. *Journal of chemical information and modeling*, 2019.

You, Y., Chen, T., Sui, Y., Chen, T., Wang, Z., and Shen, Y. Graph contrastive learning with augmentations. In *NeurIPS*, 2020.

You, Y., Chen, T., Shen, Y., and Wang, Z. Graph contrastive learning automated. In *ICML*, 2021.

A. Related Work

Graph SSL. Self-supervised learning methods for graphs are roughly categorised into contrastive and generative venues (Liu et al., 2021a;b; Wu et al., 2021; Xie et al., 2021). Contrastive graph SSL (Hu et al., 2020a; Sun et al., 2020; You et al., 2020) applies contrastive learning to maximise the mutual information between augmented instances constructed from the same graph. Generative graph SSL (Hamilton et al., 2017; Hu et al., 2020a;b; Liu et al., 2018) forms the pretext task by reconstructing original graphs. A more recent trend in Graph SSL (Liu et al., 2022; Stärk et al., 2021) is to utilise domain knowledge, e.g., 3D information of molecular conformations, to help enhance the expressiveness of GNN. In this work, we focus on studying the transferable knowledge stored in the self-supervised learned molecular graph representations.

Probing Pre-trained Embeddings. Using probe models to study learned representations is a common practice to evaluate its quality. Probe models capture the intuition that good features should perform competitively in transfer tasks even with a shallow architecture. We review the related work applying probe models for natural language processing (Conneau & Kiela, 2018; Hendricks et al., 2021; Hewitt & Manning, 2019; Jawahar et al., 2019; Kassner & Schütze, 2020; Liu et al., 2019; Tenney et al., 2019; Wang et al., 2019), computer vision (Alain & Bengio, 2017; Caron et al., 2021b; Chen et al., 2020a; 2021; He et al., 2022; Li et al., 2021; Resnick et al., 2019; Wang et al., 2021), and biomedical science (Dohan et al., 2021; Elnaggar et al., 2021; Rao et al., 2019; Rives et al., 2021; Villegas-Morcillo et al., 2021). In natural language processing, pre-trained embeddings are shown to achieve competitive results on a wide range of tasks such as token labelling and parsing. In computer vision, self-supervised learned presentations can not only improve accuracy on downstream benchmarks such as ImageNet and CIFAR10, but also contain explicit semantic information (Caron et al., 2021b). In bioinformatics and biomedical science, self-supervised learning is able to learn biological structures and functions from massive unlabelled data. It has been shown that such learned embeddings are organised at a multi-scale level and can capture the information ranging from biochemical properties of amino acids to remote homological protein structures (Rives et al., 2021).

B. Description on the Topological Property Measure

Node-level statistics focus on local topological measures of a graph, where each node is accompanied with a metric value. They could be used as features in a node classification model (Hamilton, 2020).

- **Node Degree** (d_u) counts the number of edges incident to node u: $d_u = \sum_{v \in V} \mathbf{A}[u, v]$
- **Node Centrality** (e_u) represents a node's importance, it is defined as a recurrence relation that is proportional to the average centrality of its neighbours:

$$e_u = \left(\sum_{v \in V} \mathbf{A}[u, v] e_v \right) / \lambda, \quad \forall u \in \mathcal{V} \tag{4}$$

- **Clustering Coefficient** (c_u) measures how tightly clustered a node's neighbourhood is:

$$c_u = \left(|(v_1, v_2) \in \mathcal{E} : v_1, v_2 \in \mathcal{N}(u)| \right) / d_u^2 \tag{5}$$

  *i.e.*the proportion of closed triangles in neighbourhood (Watts & Strogatz, 1998).

We use all the nodes from eight datasets, report the scores over eight test splits across multiple runs.

Graph-level statistics summarise global topology information and are helpful for tasks like graph classifications. We briefly describe their meanings and refer the formal definitions to (Hamilton, 2020).

- **Diameter**: maximum distance between the pair of nodes
- **Cycle Basis**: a set of simple cycles that forms a basis of the graph cycle space. It is a minimal set that allows every even-degree subgraph to be expressed as a symmetric difference of basis cycles.
- **Connectivity**: minimum number of elements (nodes or edges) that need to be removed to separate the remaining nodes into two or more isolated subgraphs.
- **Assortativity**: similarity of connections in the graph w.r.t the node degree, it is essentially the Pearson correlation coefficient of degree between pairs of linked nodes.

We use all the graphs from eight datasets, report the scores over eight test splits across multiple runs.

Table 5. Common classifiers trained based on the substructure counts for predicting molecular properties (ROC-AOC scores averaged over eight datasets). We utilised the conventional experimental setup in the sci-kit learn module. "Rand" and "SSL" represent the probe models trained on the randomised and GraphCL pre-trained embeddings, respectively.

Linear	RF	XGBoost	Probe (Rand)	Probe (SSL)
59.91	61.95	62.31	58.85	63.00

Pair-level statistics quantify the relationships between nodes. Since node and graph level statistics are not very useful for the tasks relied on relation modelling, we are interested in how well the pre-trained embeddings can capture the following pair-level metrics:

- **Link Prediction** tests whether two nodes are connected or not, given their embeddings and inner products. Based on the principle of *homophily*, it is expected that embeddings of connected nodes are more similar compared to disconnected pairs: $\mathbf{S}_{\text{Link}}[u, v, \mathbf{x}_u^T \mathbf{x}_v] = \mathbb{1}_{\mathcal{N}(u)}(v)$.
- **Jaccard Coefficient** seeks to quantify the overlap between neighbourhoods while minimising the biases induced by node degrees (Lü & Zhou, 2011): $\mathbf{S}_{\text{Jaccard}}[u, v] = |\mathcal{N}(u) \cap \mathcal{N}(v)| / |\mathcal{N}(u) \cup \mathcal{N}(v)|$
- **Katz Index** is a global overlap statistic, defined by the number of paths of all lengths between a pair of nodes: $\mathbf{S}_{\text{Katz}}[u, v] = \sum_{i=1}^{\infty} \beta^i \mathbf{A}^i[u, v]$, where $\beta \in \mathbb{R}^+$ is a pre-defined parameter controlling how much weight is given to short vs long paths. A small value ($\beta < 1$) down-weights the importance of long paths. Here we set $\beta = 1$, giving the paths of all lengths equal importance.

In experiments, we bootstrapped a fixed number of the node pairs (10k) from each dataset, report the test scores average over eight test splits across three runs.

C. How powerful are molecular substructure counters?

In question-answering systems, it has been found that the knowledge-aware graph modules may only carry out some simple reasoning such as counting (Wang et al., 2022). In GraphEval, we are interested in how the molecular substructure counters perform on the biochemical property predictions. We take the substructure counts as molecular descriptors to feed into classic methods, *e.g.*, linear classifier, random forest (RF), and XGBoost, which have been found (Jiang et al., 2021; Liu et al., 2018) to be effective in predicting molecular propderties.

We report the averaged test ROC-AUC scores in Table 5. Interestingly, these simple models trained on substructure counts achieve on par performance with SOTA 2D graph pre-trained embeddings. However, with more flexibility introduced by the end-to-end fine-tuning, the graph neural nets still maintain a margin of improvements ($\sim$7.7%). In retrospect to Table 1, we observe:

- with fixed pre-trained representation, GNN is comparative with substructure count descriptors + simple (linear) models;
- with fine-tuned representation, GNN perform much better than substructure counts.

Combining these two, we conjecture that GNN SSL pre-training strategies, especially contrastive-based, *e.g.*GraphCL and JOAO, are conducting something similar to substructure extraction/counting. However, it is not clear how fine-tuning pre-trained GNNs bring substantial improvements, we conjecture it might due to: (1) fine-tuning incorporate more information beyond substructure counting, such as pair/global topology; (2) GNN has larger model capacity which is born with more expressiveness.