Achieving Accurate and Explainable Drug Discovery with Zero-Cost Network Architecture Search

Harry Knighton [1] **Yiren Zhao** [1] **David Buterez** [1] **Pietro Liò** [1]

Abstract

Graph Neural Networks (GNNs) have had success in predicting the biological activity of a compound with a biological or chemical target of interest. However, as GNNs are complex and opaque, the explanation behind a prediction is not immediately apparent. To this end, we propose a method to identify highly-explainable models using Network Architecture Search (NAS) and evaluate the highest-performing models on the task of biological activity prediction. To reduce the computational cost of NAS, we leverage a range of Zero-Cost (ZC) proxies to approximate model explainability. Unfortunately, we find that existing ZC proxies underperform on this task – so we propose two novel explainability-focused ZC proxies, namely *LatentSparsityGrad* and *ConceptPurityGrad*. We demonstrate that our method performed using an ensemble of ZC proxies identifies models that are 12.9% more explainable than our baseline in only $1.49\times$ the time.

1. Introduction

Efficiently determining the activity of a compound with a biological or chemical target is vital for the drug discovery process. High-throughput screening has become widely used in both industry and academia for this purpose, with 29% of 66 published clinical drug candidates in 2016 and 2017 arising from hit compounds identified by random high throughput screening strategies (Brown & Boström, 2023). In recent years, machine learning has also had success in this task – minimising the need to expensively synthesise compounds in vitro. This has allowed for the discovery of drug candidates that function in complex or unforeseeable ways that would not be screened by a human. One example is Halicin, which although previously investigated as a diabetes drug, was found to be a powerful antibiotic using a machine learning approach (Trafton, 2020). Explaining the mechanisms behind these compounds is highly important to provide trust in a model's predictions and enable lead optimisation of hit compounds.

Graph neural networks are a good candidate for bioactivity prediction due to the complex graph structure of molecules (Buterez et al., 2022; Sakai et al., 2021). Unfortunately, due to the variety and complexity of bioassay datasets, the best-performing architectures on one dataset may not perform well on others. Thus, there is a need to identify high-performing model architectures for each dataset. Network Architecture Search (NAS) techniques have been designed specifically for graph data to achieve this (Gao et al., 2020; Zhao et al., 2020; Zhou et al., 2019).

GNNs typically have very large search spaces due to the wide range of candidate graph convolution operators available, making NAS on graph data very computationally expensive. Zero-cost proxies were introduced to avoid training models to completion in NAS and instead approximate the test loss of a model using limited data and training time (Abdelfattah et al., 2021). Abdelfattah et al. demonstrated that ZC proxies could be used to achieve the same accuracy as the previous best result on NAS-Bench-101 in a quarter of the time and further introduced the idea of using ZC proxies to warm-up the architecture search followed by a number of iterations training models to convergence.

GNNs are also not transparent about the reasoning behind why a prediction is made. As a result, methods to produce explanations for GNNs have begun appearing within the last few years (Yuan et al., 2023). One promising approach is concept-based explanation methods that cluster similar graph nodes into concepts that represent a type or quality of node (Magister et al., 2021). However, these methods require as input an existing model trained to convergence and there are currently no methods to identify highly-explainable GNN architectures. To address this, we make the following contributions:

[1]Department of Computer Science, The University of Cambridge, Cambridge, UK. Correspondence to: Harry Knighton <hjk51@cam.ac.uk>.

The 2023 ICML Workshop on Computational Biology. Baltimore, Maryland, USA, 2023.

1. We devise a method to identify explainable models using zero-cost Network Architecture Search, where model explainability is measured by adapting explainability metrics proposed in GCExplainer (Magister et al., 2021) to this task.
2. We further demonstrate that our method allows for accuracy and explainability to be jointly searched for by defining a custom objective function for NAS.
3. As existing zero-cost proxies underperform on the task of approximating explainability, we propose two novel zero-cost proxies for this purpose and construct an ensemble method to fully utilise them.

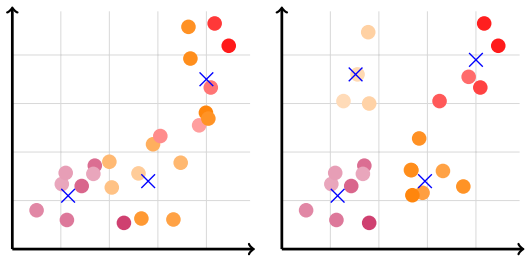
Similar approaches taken in the past have focused on explaining the search process of NAS instead of the produced models (Adam & Lorraine, 2019; Ru et al., 2021). Agiollo et al. were the first to aim for interpretable learned models by increasing the variability of local structures within the network thus reducing the complexity and opacity of the models. Zheng et al. and Carmichael et al. take this a step further by including measures of explainability in multiple objective NAS. Unlike these methods, our work considers GNNs that have a considerably larger search space than convolutional networks – inspiring our usage of ZC proxies. Further, we evaluate our method specifically on the task of bioactivity prediction for drug discovery.

2. Methodology

Measuring Explainability

GCExplainer is a concept-based method to produce global graph explanations (Magister et al., 2021). Concepts that represent a certain type or quality of node in the dataset are constructed by performing k-means clustering on the node representations in the latent space of the last graph convolutional layer in the model. An input graph can then be mapped to the concept space by encoding each node in the latent space of the model and assigning it to the nearest concept. We choose to utilise GCExplainer in this work over other global graph explainability methods for a couple of reasons: (i) GCExplainer can be computed efficiently for high dimensional biological data as it utilises k-means clustering compared to methods that rely on reinforcement learning such as XGNN (Yuan et al., 2020). (ii) As GCExplainer is an unsupervised technique, the quality of the concept clustering can be reasoned about without knowledge of the produced and ground-truth explanations.

Magister et al. evaluated GCExplainer on two metrics: concept purity and concept completeness. Concept completeness measures how much information the concept assigned to a node conveys about the label of that node. Concept purity measures how similar the concepts assigned to graphs of the same class are. In the context of drug discovery, a model with high concept completeness clusters compounds that behave similarly together in the latent space whilst allowing the representations of each graph in a cluster to differ as shown in Figure 1. Whereas, concept purity encourages latent graph representations of compounds with similar labels to be the same. This may not be ideal as this information is required for the joint optimisation of accuracy and explainability. For this reason, we use concept completeness as the measure of explainability in this work.



(a) Low Completeness (b) High Completeness

Figure 1. An illustration of how clustering in the latent space affects concept completeness.

To adapt concept completeness to a regression problem, we reformulate it as:

$$\text{Concept Incompleteness}(c) = \frac{1}{N_c} \sum_{i=1}^{N_c} (F(c) - y_i)^2$$

where c is a cluster label, N_c is the number of graphs assigned to that cluster, F is a decision tree trained to predict the label of a graph in cluster c and y_i is the label of graph i in the cluster. Mean squared error is chosen to encourage more separation between compounds with different labels as the metric increases with the square of the difference between labels. As we use this metric in NAS, we automatically select the number of clusters – the value of k in k-means – for each evaluation by choosing the value that maximises the Silhouette value.

Joint Optimisation

In order to identify models that are both accurate and explainable, we define a joint objective function for NAS:

$$\mathcal{L}(M) = \delta \cdot \text{Acc}(X, y; M) + (1 - \delta) \cdot \text{Explain}(X, y; M)$$

where X and y are the features and labels of the test data, M is the model architecture to be evaluated, and 'Acc' and 'Explain' are measures of – or proxies for – accuracy and explainability respectively for a model M. The parameter $\delta \in [0, 1]$ is set by the user to represent whether NAS should optimise more heavily for explainability or accuracy. Values closer to 0 indicate a preference for explainability and values

closer to 1 indicate a preference for accuracy. This design allows the user to fine-tune the search to the requirements of their work by tweaking the δ parameter.

Zero-cost Proxies

We utilise eight existing ZC proxies in this work and propose two novel ones. Of the eight existing ZC proxies, two are data-dependent: NumParams and SynFlow, and six are data-independent: Snip, Grasp, Fisher, JacobCov, GradNorm and ZiCo (Abdelfattah et al., 2021). Experimentally, we find that the existing ZC proxies all have a low correlation with concept completeness. This suggests the need for new proxies designed to measure explainability. Thus, we propose two novel explainability-oriented ZC proxies:

LatentSparsityGrad — We define this gradient-based ZC proxy to assign a saliency score to each parameter of

$$\text{LatentSparsityGrad}(\theta_i) = \left| \frac{\delta \text{PD}(\mathbf{x})}{\delta \theta_i} \right|$$

where PD calculates the average pairwise distance between representations in the latent space of the model. The reasoning behind this definition is that parameters that cause a large amount of movement in the latent space of the model contribute more to clustering similar data points.

ConceptPurityGrad — Similarly, we define another ZC proxy using concept purity (CP) from Magister et al.:

$$\text{ConceptPurityGrad}(\theta_i) = \left| \frac{\delta \text{CP}(\mathbf{x}, \mathbf{y})}{\delta \theta_i} \right|$$

We choose to use a linear regression model to combine the information from each proxy for the following reasons: (i) Linear regression models are computationally inexpensive to train. (ii) There are significant linear correlations between proxies and metrics in most cases. (iii) The coefficients in a linear regression model can adjust for the existence of negative and positive correlations between proxies.

In order to fit the linear model, some number of GNNs will need to be trained to convergence. We want to minimise the number of GNNss trained whilst retaining enough information to sufficiently fit the model. From Figure 2, it is clear that a greater number of samples is needed than the number of proxies used in the ensemble as below this the system is underdetermined. Sampling 30 models brings the mean-squared error to below 0.008, which decreases by only 0.003 after another 70 samples are added.

3. Experiments

Our implementation is publicly available on GitHub.

Data — We evaluate our methods on 7 bioassay datasets curated from PubChem data (Kim et al., 2023) published in

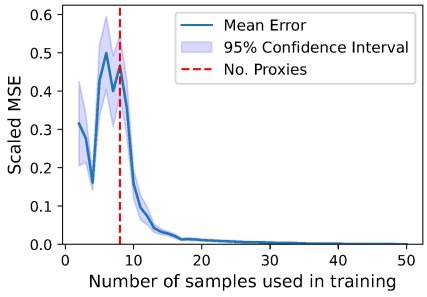


Figure 2. A graph of the test loss of the ensemble of zero-cost proxies against the number of training samples used to fit the ensemble. Training data is collected by evaluating zero-cost proxies and target metrics on randomly sampled architectures.

(Buterez et al., 2023): AID1445, AID504329, AID624330, AID1465, AID449756-435005, AID1431-873, AID504313-2732. These datasets are chosen to represent a variety of methodologies and measurements used in the collection. Each dataset contains from 569 to 1811 high-quality dose-response measurements of compounds. The dose-response data for each dataset is split in 5 ways, each with 70% of the data for training, 20% for validation and 10% for testing.

Baseline — We construct baseline results by recreating those published in (Buterez et al., 2022). We hand-design twelve baseline GNN architectures each using one of 4 graph convolution layer types: GCN, GIN, GAT and GATv2, and one of 3 global pooling methods: sum, mean and maximum. A fixed architecture of linear readout layers is appended after the graph convolutional layers of each model. Results are averaged over the five dataset splits and three random initialisation seeds.

Experimental Set-up — We constrain the search space by limiting GNN architectures to at most 3 graph convolution layers as the number of parameters in the search space is exponential in the number of layers and architectures with many convolutional layers suffer from the over-smoothing phenomenon (Chen et al., 2020). Further, we set all convolutional layers to the same width and use the same fixed readout architecture as used in the baseline models. We warm up NAS through 10000 evaluations using random search with an ensemble of ZC proxies. This is followed by 100 evaluations using the Tree-Structured Parzen Estimator (TPE) algorithm and training each model to convergence to calculate the test loss. Random search is chosen to explore the search space using the ZC proxies and the main search is performed using TPE to better exploit the information gained from the warm-up. We consider two ensemble methods: one using only the existing ZC proxies and one using the existing ZC proxies and our novel ZC proxies. The ensembles utilise a single minibatch of 64 data points to calculate each ZC proxy and the linear model is fit using

the results from 50 randomly sampled architectures for each dataset. Results for the highest-performing architecture on each dataset are reported using the same evaluation strategy as the baseline experiments.

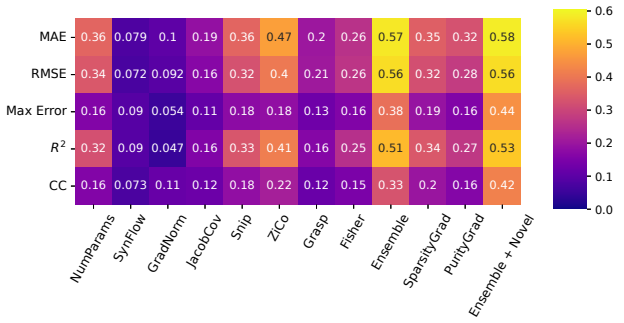


Figure 3. Heatmap of magnitudes of Pearson correlations between zero-cost proxies and target accuracy and explainability metrics over 7 bioassay datasets. Metrics are listed along the left and proxies along the bottom. Concept completeness is labelled as CC. High correlation is desired as the relative ordering of labels and predictions is important. We see that the ensemble method including our novel zero-cost proxies performs the best on all metrics and is significantly more correlated with explainability than the ensemble of only existing proxies. The highest p-value of the correlation of the novel ensemble is 5.28×10^{-2} with RMSE and 2.01×10^{-3} with concept completeness.

Results — The ensemble method of existing proxies significantly outperforms all of the standard proxies. Most notably, the ensemble has a 17.5% higher correlation with RMSE and 33.3% higher correlation with concept completeness than *ZiCo*. With regards to our novel proxies, *LatentSparsityGrad* has a higher correlation with all metrics than *ConceptPurityGrad* – although *ConceptPurityGrad* is still useful as each conveys different information. *LatentSparsityGrad* is the second most correlated proxy with concept completeness after *ZiCo* scoring only 0.02 below. Further, *LatentSparsityGrad* is competitive with the next two highest scoring proxies: *NumParams* and *Snip*. The most promising result of the novel proxies arises when they are combined into an ensemble with the existing proxies: when included, the correlation with concept completeness increased by 27.3% from 0.33 with the ensemble of existing ZC proxies to 0.42.

The ensemble using only existing ZC proxies performs 13.5% worse than our baseline in $1.59\times$ more time. When incorporating the novel proxies into the ensemble, the ensemble performed 12.9% better than our baseline in $1.49\times$ the time and 23.3% better than the previous ensemble.

In Table 2, we observe that, as expected, high values of δ lead to better R^2 and max error values and lower values of δ lead to better concept completeness values. With 90% confidence, R^2 and concept incompleteness have a Pearson correlation of 0.265. However, notice that concept incompleteness decreases and R^2 increases when lowering δ from 1.0 to 0.5. This is due to the effect of concept completeness on the latent space of the model. Optimising for concept completeness encourages graphs with different labels to be further apart in the latent space whilst ensuring that graphs with similar labels do not converge to a single representation. This allows the readout layers to better predict the ordering of similar compounds, so R^2 increases.

Table 1. Run times and concept incompleteness scores over 7 bioassay datasets for our baseline results, NAS using the ensemble of existing proxies and NAS using the ensemble of existing ZC proxies and our novel ZC proxies. Lower run time and concept incompleteness is desired and the lowest of each is given in bold. Our method using the novel proxies can identify models that are 12.9% more explainable than our baseline in $1.49\times$ the time.

Experiment	Run Time (s)	Conc. Incompleteness
Baseline	$\mathbf{2172} \pm 268$	0.155 ± 0.00294
Ensemble	3447 ± 1038	0.176 ± 0.0502
Ensemble + Novel ZC Proxies	3226 ± 1097	$\mathbf{0.135} \pm 0.0657$

Table 2. R^2, max error and concept incompleteness of the optimal architecture identified by NAS when optimising for differing proportions of loss and explainability over the 7 datasets. Low δ indicates a preference for explainability and high δ for accuracy. Higher R^2 and lower max error and concept completeness are optimal. The best value for each metric is given in bold.

Loss-Explainability (δ)	R^2	Max Error	Conc. Incompleteness
Baseline	0.234 ± 0.0106	1.27 ± 0.0339	0.155 ± 0.00294
0	0.259 ± 0.162	1.36 ± 0.41	$\mathbf{0.124} \pm 0.0678$
0.25	0.335 ± 0.174	$\mathbf{1.00} \pm 0.269$	0.129 ± 0.0711
0.5	0.455 ± 0.203	1.05 ± 0.31	0.131 ± 0.0699
0.75	$\mathbf{0.459} \pm 0.138$	1.01 ± 0.364	0.125 ± 0.0518
1.0	0.367 ± 0.154	1.10 ± 0.332	0.167 ± 0.0646

4. Discussion and Conclusions

In this paper, we proposed a novel zero-cost NAS-based method for identifying explainable and accurate models using zero-cost proxies. Our method demonstrates that zero-cost NAS can be used to identify highly explainable models. However, in this work, a high explainability score only indicates that a model can identify structural similarities between compounds and cluster them accordingly. In addition to this, we recommend that explanations for hit compounds should be generated and manually evaluated to ensure that explanations align with the user's domain knowledge of the task. To determine whether concept completeness is representative of the quality of explanations produced by a model, further research would be required over a range of tasks guided by domain experts.

References

Abdelfattah, M. S., Mehrotra, A., Dudziak, L., and Lane, N. D. Zero-cost proxies for lightweight NAS. 2021. URL http://arxiv.org/abs/2101.08134.

Adam, G. and Lorraine, J. Understanding neural architecture search techniques, 2019.

Agiollo, A., Ciatto, G., and Omicini, A. *Shallow2Deep: Restraining Neural Networks Opacity Through Neural Architecture Search*, pp. 63–82. 07 2021. doi: 10.1007/978-3-030-82017-6_5.

Brown, D. G. and Boström, J. Where do recent small molecule clinical development candidates come from? *Journal of Medicinal Chemistry*, 61(21):9442–9468, 2023. ISSN 0022-2623. doi: 10.1021/acs.jmedchem.8b00675. URL https://doi.org/10.1021/acs.jmedchem.8b00675. Publisher: American Chemical Society.

Buterez, D., Janet, J. P., Kiddle, S. J., and Lio, P. Multi-fidelity machine learning models for improved high-throughput screening predictions. 2022. URL http://doi.org/10.26434/chemrxiv-2022-dsbm5-v2.

Buterez, D., Janet, J. P., Kiddle, S. J., and Liò, P. MF-PCBA: Multifidelity high-throughput screening benchmarks for drug discovery and machine learning. *Journal of Chemical Information and Modeling*, 2023. ISSN 1549-9596. doi: 10.1021/acs.jcim.2c01569. URL https://doi.org/10.1021/acs.jcim.2c01569. Publisher: American Chemical Society.

Carmichael, Z., Moon, T., and Jacobs, S. A. Learning interpretable models through multi-objective neural architecture search. *ArXiv*, abs/2112.08645, 2021.

Chen, D., Lin, Y., Li, W., Li, P., Zhou, J., and Sun, X. Measuring and relieving the over-smoothing problem for graph neural networks from the topological view. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):3438–3445, 4 2020. doi: 10.1609/aaai.v34i04.5747. URL https://ojs.aaai.org/index.php/AAAI/article/view/5747.

Gao, Y., Yang, H., Zhang, P., Zhou, C., and Hu, Y. Graph neural architecture search. In Bessiere, C. (ed.), *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pp. 1403–1409. International Joint Conferences on Artificial Intelligence Organization, 7 2020. doi: 10.24963/ijcai.2020/195. URL https://doi.org/10.24963/ijcai.2020/195. Main track.

Kim, S., Chen, J., Cheng, T., Gindulyte, A., He, J., He, S., Li, Q., Shoemaker, B. A., Thiessen, P. A., Yu, B., Zaslavsky, L., Zhang, J., and Bolton, E. E. PubChem 2023 update. *Nucleic Acids Research*, 51:D1373–D1380, 2023. ISSN 1362-4962. doi: 10.1093/nar/gkac956.

Magister, L. C., Kazhdan, D., Singh, V., and Liò, P. GCExplainer: Human-in-the-loop concept-based explanations for graph neural networks, 2021. URL http://arxiv.org/abs/2107.11889.

Ru, B., Wan, X., Dong, X., and Osborne, M. Interpretable neural architecture search via bayesian optimisation with weisfeiler-lehman kernels, 2021.

Sakai, M., Nagayasu, K., Shibui, N., Andoh, C., Takayama, K., Shirakawa, H., and Kaneko, S. Prediction of pharmacological activities from chemical structures with graph convolutional neural networks. *Scientific Reports*, 11(1):525, 2021. ISSN 2045-2322. doi: 10.1038/s41598-020-80113-7.

Trafton, A. Artificial intelligence yields new antibiotic, 2020. URL https://news.mit.edu/2020/artificial-intelligence-identifies-new-antibiotic-0220.

Yuan, H., Tang, J., Hu, X., and Ji, S. Xgnn: Towards model-level explanations of graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, pp. 430–438, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450379984. doi: 10.1145/3394486.3403085. URL https://doi.org/10.1145/3394486.3403085.

Yuan, H., Yu, H., Gui, S., and Ji, S. Explainability in graph neural networks: A taxonomic survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5782–5799, 2023. doi: 10.1109/TPAMI.2022.3204236.

Zhao, Y., Wang, D., Gao, X., Mullins, R., Lio, P., and Jamnik, M. Probabilistic dual network architecture search on graphs, 2020. URL http://arxiv.org/abs/2003.09676.

Zheng, X., Wang, P., Wang, Q., Shi, Z., and Fan, J. Disentangled neural architecture search. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2022. doi: 10.1109/IJCNN55064.2022.9892784.

Zhou, K., Song, Q., Huang, X., and Hu, X. Auto-gnn: Neural architecture search of graph neural networks. *arXiv preprint arXiv:1909.03184*, 2019.