Improving generalisability of 3D binding affinity models in low data regimes

Julia Buhmann*
Exscientia
jbuhmann@exscientia.co.uk

Ward Haddadin*
Exscientia
whaddadin@exscientia.co.uk

Lukáš Pravda
Exscientia
lpravda@exscientia.co.uk

Alan Bilsland
Exscientia
abilsland@exscientia.co.uk

Hagen Triendl
Exscientia
hagentriendl@gmail.com

Abstract

Predicting protein-ligand binding affinity is an essential part of computer-aided drug design. However, generalisable and performant global binding affinity models remain elusive, particularly in low data regimes. Despite the evolution of model architectures, current benchmarks are not well-suited to probe the generalisability of 3D binding affinity models. Furthermore, 3D global architectures such as GNNs have not lived up to performance expectations. To investigate these issues, we introduce a novel split of the PDBBind dataset, minimizing similarity leakage between train and test sets and allowing for a fair and direct comparison between various model architectures. On this low similarity split, we demonstrate that, in general, 3D global models are superior to protein-specific local models in low data regimes. We also demonstrate that the performance of GNNs benefits from three novel contributions: supervised pre-training via quantum mechanical data, unsupervised pre-training via small molecule diffusion, and explicitly modeling hydrogen atoms in the input graph. We believe that this work introduces promising new approaches to unlock the potential of GNN architectures for binding affinity modelling.

1 Introduction

Computer-aided drug design relies on the accurate prediction of protein-ligand binding affinity to achieve a therapeutic effect, ensuring selectivity against other proteins and avoiding off-target toxicity [Kairys et al., 2019].

Apart from classical approaches for binding affinity prediction (usually docking methods which use a combination of empirical molecular and statistical force fields [Jones et al., 1997, Eberhardt et al., 2021]), a diverse array of machine learning (ML) strategies have been proposed in the last decade. There has been increasing interest in developing ML models that use 3D data of protein-ligand complexes as input. In principle, these models are the best suited for predicting binding affinity since they should be able to capture fundamental interaction mechanisms such as hydrogen bonds

*Equal contribution.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

or hydrophobic and ionic interactions between the protein and the ligand. Numerous types of 3D ML models are presented in the literature. In Ballester and Mitchell [2010] and Wang et al. [2021], the protein and ligand interactions were condensed into contact map features and used in tree-based models (Random Forest) and convolutional neural networks (CNN). In Volkov et al. [2022], they use a graph neural network (GNN) which uses the protein-ligand graph as input. They also propose including interacting nodes in the graph to indicate known interactions between protein and ligand atoms. Using GNNs as well, Zhang et al. [2023] add thresholds to the distance encoding to avoid overfitting on small distance variations.

The majority of the literature indicates that it is not yet clear whether a specific model type consistently achieves the best results [Durant et al., 2023]. Part of this ambiguity is due to the lack of consistent benchmarks to evaluate the performance of the diverse array of models. The common dataset used for benchmarking 3D binding affinity models is the PDBBind [Liu et al., 2015], a dataset of crystal structures from the Protein Data Bank (PDB) with curated binding affinity measurements. Over the last few years, many splits have been proposed for the PDBBind dataset to probe performance and generalisation of different model types [Volkov et al., 2022, Durant et al., 2023, Li et al., 2024], but each come with their own drawbacks.

Overall, the results indicate that simple models and baselines perform just as well as the more complicated 3D models that use structural information [Durant et al., 2023]. This indicates that 3D models are not learning generalizable information but only dataset biases, hence they have not yet met their projected expectations [Volkov et al., 2022].

In this work we investigate the performance of binding affinity model families in a robust setting to probe what they learn and how they generalise. We compare 3D global models to protein-specific local models commonly used in real world drug discovery and also to baseline bias models. To achieve this, we propose a new split of the PDBBind dataset based on protein and ligand similarity and constructed to suit bechmarking the various model families fairly and consistently. We use the new split and strong baselines to test multiple novel improvements to a plain 3D GNN model to push the boundaries of binding affinity modelling using 3D GNNs.

We find that in low data regimes, 3D models significantly outperform protein-specific local models. With more data for a specific protein, local models quickly catch up. We also investigate the effect of hydrogen atoms on generalisability. As the structures in the PDBBind are not consistently prepared, we use protein preparation software to prepare them consistently and include hydrogen atoms explicitly in the GNN encoding. We again find that at low data regimes, including hydrogen atoms explicitly is very important for generalisation. This advantage goes away with more data. Finally, we propose two pre-training methods to improve global 3D model performance. We pre-train GNNs on supervised quantum mechanical energy prediction and unsupervised small molecule diffusion. We show that both result in improvements at low data regimes. As far as we are aware, this is the first application of quantum mechanical pre-training and diffusion pre-training for binding affinity prediction.

2 Methods

In this section, we present an overview of our methods and benchmarks. We discuss the dataset, structure preparation, model families benchmarked, and the new proposed split. We have made the code, prepared structures, and splits in this work publicly available at `https://github.com/Exscientia/low-sim-pdbbind` and `https://zenodo.org/records/13772124`.

2.1 PDBBind dataset

We choose the PDBBind dataset (release v2020[2]) as our benchmarking data set [Liu et al., 2015]. The data consists of crystal structures of bound protein-ligand complexes deposited in the PDB with curated binding affinity values (K_I, K_D or IC_{50}). We use the protein-ligand subset of the general set from PDBBind consisting of 19443 unique protein-ligand structures. The dataset is unbalanced. Many proteins have binding affinities measured only against a single ligand (one structure), some

[2]As of 2024, there is a newer PDBBind release (v2021) available at `https://www.pdbbind-plus.org.cn/`. We did not use it in this work, as access is restricted.

have measurements for a few ligands (few structures), and very few have measurements against more than 100 ligands.

It has been demonstrated, in both predictive and generative settings [Durant et al., 2023, Li et al., 2024, Buttenschoen et al., 2023], that the splits routinely used in model benchmarking on PDBBind contain data leakage. The similarity between proteins and ligands across training and test sets inflates metrics for certain models or tasks and makes rigorously probing their performance and generalisability difficult.

2.2 Structure preparation

We queried the PDB [wwPDB consortium, 2019] for biological assemblies of all the structures listed in the PDBBind dataset. Due to the nature of structure-determining techniques, the structures contain not only ligands of biological nature, but also residues of crystalization buffers or cryoprotectants. Hydrogen atoms are also mostly missing.

To address both of these issues, each structure was prepared using CCDC software [Groom et al., 2016], namely the Python API (`v.3.0.16`). Hydrogen atoms were added and the ligands listed by PDBBind as biologically relevant were extracted into separate files. The remaining ligands of non-biological nature along with water molecules were removed.

After preparation, 18310 structures out of the total 19443 remained (1133 failed).

2.3 Models

Next, we discuss the models used in this study. An overview is presented in Table 1. A full description of the hyperparameters of the features and models is provided in the Supplementary Information.

Table 1: Overview of the different models used in this study. We indicate whether the models are global or local and whether they use 3D information or not. We make use of two baseline models to estimate dataset biases (Ligand-Bias and Molecular-Weight).

Model Name	Global/Local	3D/non-3D	Input
Single-Protein	local	non-3D	ligand only
EGNN [Satorras et al., 2022]	global	3D	pocket + ligand
RF-Score [Ballester and Mitchell, 2010]	global	3D	pocket + ligand
OnionNet-2 [Wang et al., 2021]	global	3D	pocket + ligand
Ligand-Bias	global	non-3D	ligand only
Molecular-Weight	local	non-3D	ligand only

2.3.1 Model families

Before presenting the models used, we establish clear groups to classify them. There are numerous ways to categorize binding affinity model types, but in this work we introduce a particular grouping focused on two key aspects: the scope of application, global vs. local, and the type of data input, non-3D vs. 3D.

We denote models intended for use on different protein targets by **global models** and ones intended for use against a single protein target by **local models**. Local models are typically trained on ligand activity data measured against a single protein. Binding affinity models can be further grouped according to how ligands and proteins are represented. We denote models which do not use 3D coordinates by **non-3D models**. These models often use ligand descriptors such as fingerprints to encode ligands. If protein information is used, it is most commonly encoded with its amino acid sequence. **3D models** use the 3D coordinates of a ligand or a protein-ligand complex alongside non-3D information.

2.3.2 Single-Protein local models

To compare to standard practices in the drug discovery community and in real world projects [Lo et al., 2018], we benchmark the performance of Single-Protein local models built from ligand-based features

and trained on binding activity data measured against a single protein. Analogous to established workflows [Jiang et al., 2021, Deng et al., 2023], we use classical ML models (Random Forest [Breiman, 2001], XGBoost [Chen and Guestrin, 2016], CatBoost [Prokhorenkova et al., 2018], Support Vector Machine [Cortes, 1995]) combined with a selection of fingerprints (ECFP, FCFP, atom pairs, topological torsion) and molecular descriptors for features. More details on features and models are listed in the Supplementary Information.

2.3.3 EGNN models

We use the EGNN [Satorras et al., 2022] model[3] as the base architecture for our 3D global models. We use the implementation in the PHYSICSML python package Exscientia [2024b]. As in previous work, we extract the protein pocket by selecting protein atoms within 5Å of any ligand atom. The graph is constructed from the pocket and ligand atoms as nodes and a 5Å cut-off is used to define the edges. We use one-hot encoded atomic numbers as node features and no edge features.

Pre-Trained EGNNs In this benchmark, we use a few different versions of the EGNN model. In addition to the basic model, we use two pre-trained versions. The first model, called **EGNN-QM**, is trained on ANI1ccx, a dataset of 500k small molecules with quantum mechanical energies computed at the ccsd(t) level. The knowledge learned from quantum mechanical interactions and internal energies should in principle better inform binding affinity prediction. The second model, **EGNN-DIFF**, is trained as a small molecule diffusion model on the QM9 dataset (as described in Hoogeboom et al. [2022]). By pre-training on diffusing stable QM9 molecules, the model will have learned to distinguish between low and high energy conformations. In principle, this should allow it to better understand binding affinity interactions.

To transfer these models to the PDBBind dataset, we use a two stage procedure. First, we freeze the backbone and add a new randomly initialised pooling head and train until convergence. Then, we unfreeze the backbone and train all parameters at a lower learning rate until convergence.

Information about both pre-training strategies and transfer learning is available in the Supplementary Information.

Hydrogens Previous works modelling binding affinity via GNNs have chosen to omit hydrogen atoms from the input graph (Li et al. [2021], Volkov et al. [2022]). Since hydrogen atoms contribute significantly to binding via hydrogen bonds, we wanted to assess the effect of including hydrogen atoms as nodes in the graph. We benchmark the models with no hydrogen atoms (**None**), with only the polar hydrogen atoms (**Polar**), and with all hydrogens (**Explicit**).

Single-Graph vs. Multi-Graph Finally, to probe whether the models learn to identify the interactions from the protein-ligand pose, we also train models on the pockets and ligands as separate graphs. We use the same backbone to generate embeddings for both and then combine these embeddings in a pooling head to make the final prediction. Practically, this removes any edges between protein and ligand nodes in the graph. We refer to these models, which treat proteins and ligands as separate graphs, as **Multi-Graph** models. On the other hand, the conventional models that operate on the interacting pose are referred to as **Single-Graph** models.

2.3.4 RF-Score and OnionNet-2

We include in our analysis two additional 3D global models, RF-Score and OnionNet-2. We select those models due to their performance on other splits of the PDBBind dataset as indicated in the study by Durant et al. [2023]. Specifically, RF-Score was one of the top performer on the CASF-2016 split while the OnionNet-2 model was superior on the 2019-Holdout and Peptides-Holdout sets relative to other models tested.

[3]Although not included in this work, we also benchmarked more advanced 3D models such as MACE (Batatia et al. [2023]), Allegro (Musaelian et al. [2022]), and NequIP Batzner et al. [2022]). However, we saw very poor performance. These models are top performers for large high quality datasets like quantum mechanincal energy prediction. We hypothesise that the reason for poor performance in this context is due to the large number of parameters and complex interactions which are more susceptible to overfitting on a biased dataset like PDBBind.

2.3.5 Baseline models

Ligand-Bias model To probe the ligand dataset bias in the benchmarking splits, we follow the work of Durant et al. [2023] and design a Ligand-Bias global model. This model is trained on the identical data used to train the global models (binding affinity measurements of ligands against different proteins), but only uses ligand-based features as input (analogous to Single-Protein models; see 2.3.2). Effectively, this mixes binding affinity values of compounds measured against different proteins to probe the amount of dataset bias available in ligand information alone.

Molecular-Weight model The molecular weight of a compound tends to be a strong predictor of its binding affinity, with larger compounds generally exhibiting stronger affinity [Olsson et al., 2008]. Within the context of drug discovery, it is particularly important to avoid building binding affinity models that strongly make use of the molecular weight property, as larger drug candidates have a higher probability of failure [Hopkins et al., 2014]. In this study, we employ a Molecular-Weight model that uses the molecular weight of the ligand as its sole input as a baseline. We use the same architectures and training data as for the Single-Protein local models.

2.4 Splitting

To build a robust benchmark and effectively probe model generalisation, we propose a new split of PDBBind based on protein and ligand similarity, which we call the **Low-Sim** split. Although many splits have been proposed, we believe that none achieve our goal of probing generalisation. Close inspection of the proposed splits [Volkov et al., 2022, Durant et al., 2023, Li et al., 2024] shows non-negligible levels of similarity between train and test set, with all splits sharing some proteins (UniProts) across sets. Table 2 shows the amount of UniProt overlap in previously proposed splits. Furthermore, the splits were not constructed to benchmark the variety of model families available (local vs. global, 3D vs. non-3D).

Table 2: Overview of UniProt overlap in previously proposed splits. The numbers denote the number of unique overlapping UniProts, overlapping test structures (out of total test structures), and overlapping train structures (out of total train structures).

Split name	# UniProts overlap	# train overlap	# test overlap
Post 2019 set [Volkov et al., 2022]	262	5520 / 16561	1004 / 1467
CASF 2016 [Volkov et al., 2022]	67	3944 / 16561	281 / 282
Zero-Ligand-Bias [Durant et al., 2023]	170	3724 / 17605	287 / 360
LeakProof PDBBind [Li et al., 2024]	172	1735 / 12923	3526 / 4751
Low-Sim (Ours)	0	0 / 11022	0 / 1857

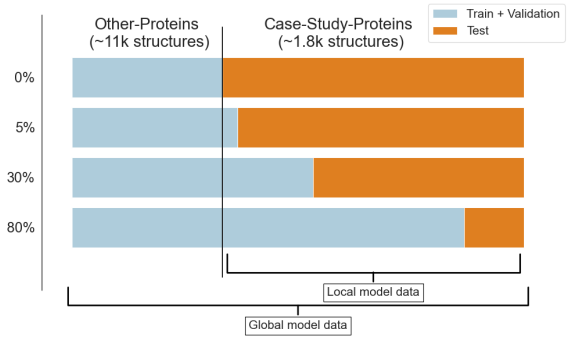
Case-Study-Proteins We design our newly proposed dataset splits such that we can compare between global and local models. This comparison is particularly relevant given that local Single-Protein models are still widely used in drug discovery projects, owing to their robustness, cost-effectiveness, and trainability on smaller datasets. To obtain a direct and fair comparison to these local models, we select eight proteins from the PDBBind dataset which have more than 100 datapoints as our case study. The threshold of 100 points per protein is to allow enough data to train local models. In total, the Case-Study-Proteins consists of 1857 structures. These proteins will be used to benchmark the generalisability of global models and also to train local models for each specific protein.

Similarity filtering We apply two similarity filtering steps to the remaining structures in the PDBBind dataset to create a subset that is dissimilar to the Case-Study-Proteins set. We call this reduced dataset Other-Proteins.

To account for protein similarity, we compute the similarity to the Case-Study-Proteins using FoldSeek [van Kempen et al., 2024], which uses 3D structural and residue information to efficiently compute a similarity score [0, 1] between two protein structures. To probe the generalisability of the models, we filter out any structures which have more than 0.5 similarity to the Case-Study-Proteins structures. Additionally, we compute the tanimoto similarity between the Case-Study-Proteins ligands and the

Table 3: Case-Study-Proteins

UniProt	HGNC	Number of structures
P00734	F2	170
P56817	BACE1	343
P24941	CDK2	248
O60885	BRD4	199
P00918	CA2	425
P07900	HSP90AA1	172
Q9H2K2	TNKS2	113
P00760	N/A (Bovine)	187



Dataset	# structures
PDBBind	19443
PDBBind prepared	18310
Low-Sim	12879
- Case-Study-Proteins	1857
- Other-Proteins	11022

Figure 1: Schematic of Low-Sim benchmarking splits used in this study. Global models are trained on both train sets from the Case-Study-Proteins and the Other-Proteins. Local models are individually built for each of the eight proteins in the Case-Study-Proteins split. They require a minimum set of already available ligands for a specific protein for training, thus can only be created for the 5%, 30%, 80% splits and use only data from the Case-Study-Proteins split. Note that bars are not to scale with number of samples.

ligands of the remaining structures. We filter out any structures with ligands with more than 0.5 tanimoto similarity. A total of 5431 similar structures are removed and leaves 11022 structures in the Other-Proteins set.

Low-Sim 0%, 5%, 30%, 80% We aim to examine the change in model performance as the amount of data increases, from low data regimes (when binding affinity values are available for approximately 0 to 30 ligands for a specific protein) in comparison to a medium data scenario where increasingly more data is available.

We use the Low-Sim split (1875 Case-Study-Proteins structures and 11022 Other-Proteins structures) to construct the benchmarking splits as follows. We stratify the Case-Study-Proteins data by protein and split the structures by tanimoto ligand similarity[4] with increasing fraction of training data, 5%, 30%, 80%. At each percentage, we generate three folds with a different starting seed ligand for the similarity splits. Local models (Single-Protein and Molecular-Weight) are trained on these splits for each protein individually. For the global models (RF-Score, OnionNet-2, EGNN, Ligand-Bias baseline), we further augment the train sets of these splits with the Other-Proteins. Additionally, we construct the 0% split where all Case-Study-Proteins are in the test set and only the Other-Proteins are in the train set. This is to probe generalisation to completely new proteins. The final benchmarking splits are shown schematically in Figure 1. In absolute numbers, those splits correspond to the following number of training samples per protein: 11±5 (5%), 69±30 (30%), 185±83 (80%).

[4]We note that tanimoto splits are harder than scaffold splits since scaffold which technically are distinct can still have high tanimoto similarity.

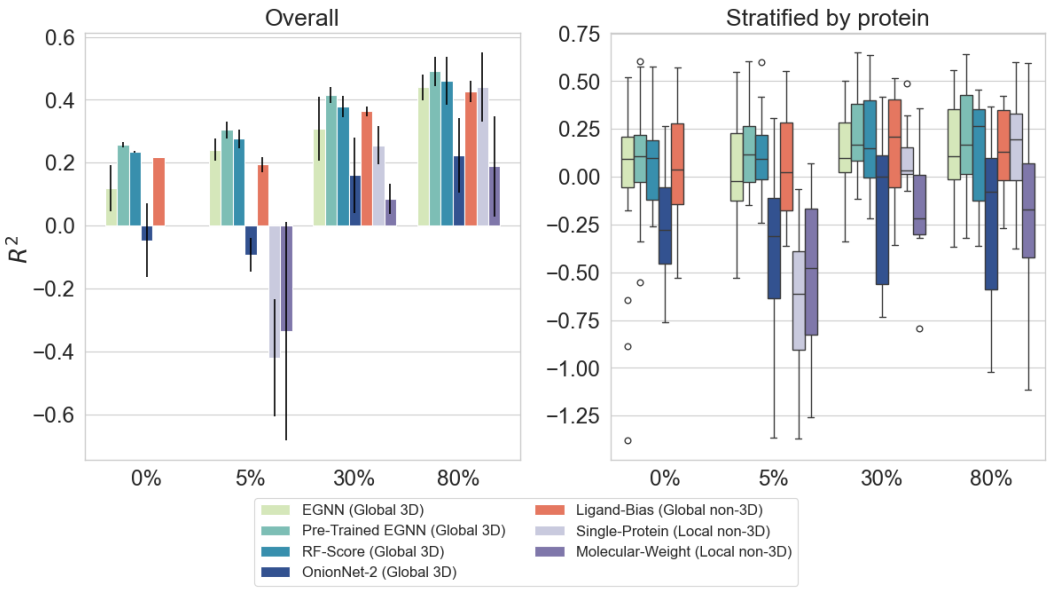


Figure 2: Overall and stratified performance at increasing train data fraction for different model families. In the low data regime, global 3D models outperform local models. Left: The error bars denote the standard deviation across the three test folds. Right: The boxplots represent the performance distribution over the eight proteins in the Case-Study-Proteins set.

2.4.1 Training and validation

For each split (0%, 5%, 30%, and 80%), we perform a cross-validation split of (random 80:20 split) for model selection. The models are then retrained on the combined train and validation set and the performance is measured on the test set.

For deep learning models (EGNN architectures), we additionally use a random 80:20 split of the train set for early stopping.

2.4.2 Metrics

Typically, binding affinity models are assessed using the Pearson correlation coefficient, which measures the correlation between predicted and actual binding affinity values. However, this doesn't measure the absolute predictive performance, which is crucial in real-world drug discovery for optimising multi-parameter objectives. Therefore, here we focus on the absolute R^2 metric for benchmarking performance, while also providing root mean squared error and Pearson correlation results in the Supplementary Information.

On the Low-Sim test sets, we compute the metrics in two different ways. The **overall** performance refers to metrics computed on the predicted and actual binding affinity values across all eight proteins. Mixing predictions of all the different protein-ligand pairs is the common approach when reporting results on the PDBBind dataset [Meli et al., 2022]. We also report the performance **stratified by protein**, where metrics are calculated individually for each protein.

3 Results

We now present the results of the benchmarking. First, we present the results and comparisons across model families. We follow this with more detailed analyses of the 3D EGNN models.

3.1 Model family comparisons

The results reveal a clear advantage for global 3D models over local models in low data regimes. As seen in Figure 2, the global 3D models (EGNN, pre-trained EGNN, and RF-Score) have moderate

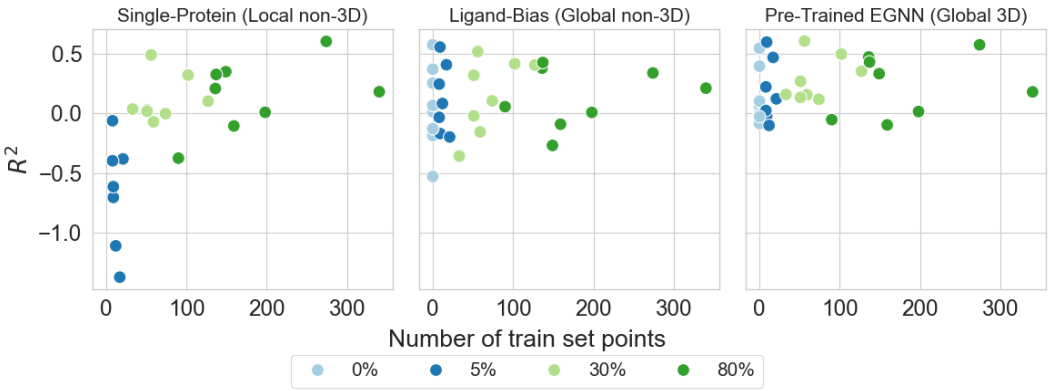


Figure 3: Effect of number of training data points on performance. Each point represents a protein from the eight case-study proteins. The global models show a clear advantage at low data regimes.

generalisation even with 0% protein-specific training data and outperform the Single-Protein model at lower train data fractions (5% and 30%). With enough data (80%), most models plateau at similar performance (EGNN, pre-trained EGNN, RF-Score, Single-Protein). We hypothesise that this is due to the difficulty of the tanimoto similarity splits. At low data levels, the local models not only have very little data to learn from but also a train set which is very dissimilar to the test ligands. In drug discovery settings where generalisation is important, this benchmark clearly highlights the importance of global models.

Notably, the pre-trained EGNN models outperform all other models and baselines when considering the overall performance (Figure 2, left). See Section 3.2 for a detailed analysis of their performance.

One thing to note is the relatively good performance of the Ligand-Bias model (red bar in Figure 2). Although not as good as the global 3D models, the Ligand-Bias model is able to generalise to the unseen proteins. We hypothesise that this is due to the model picking up on functional group importance in the ligands, given that a lot of binders rely on generic functional groups.

It is also interesting to note that, as seen in Figure 3, the performance of the global 3D models is relatively consistent with the level of training data (improving slightly) whereas the Single-Protein local model suffers greatly at low data, dramatically improving with increasing amounts of training data.

3.2 Improvements to EGNNs

We now present a detailed analysis of the global 3D EGNN models.

3.2.1 Pre-Training

First, we look at the effect of pre-training on model performance. As Figure 4 shows, pre-training significantly improves model performance. Quantum mechanical pre-training provides the largest advantage followed closely by diffusion pre-training. As expected, the advantages are more noticeable at low levels of training data, gradually fading out as more training data is added. To our knowledge, this is the first application of using pre-trained models for 3D binding affinity prediction.

3.2.2 Hydrogens

Next, we assess the effect of including hydrogen atoms on model performance. To our knowledge, in all previously published work, hydrogen atoms were ignored in the structures. It was not clear if this occurred due to better model performance on the benchmarks or due to knowledge of the poorly prepared PDBBind structures. As described in section 2.2, we noticed that the structures in PDBBind did not have consistent hydrogen preparation and used CCDC software to add them consistently.

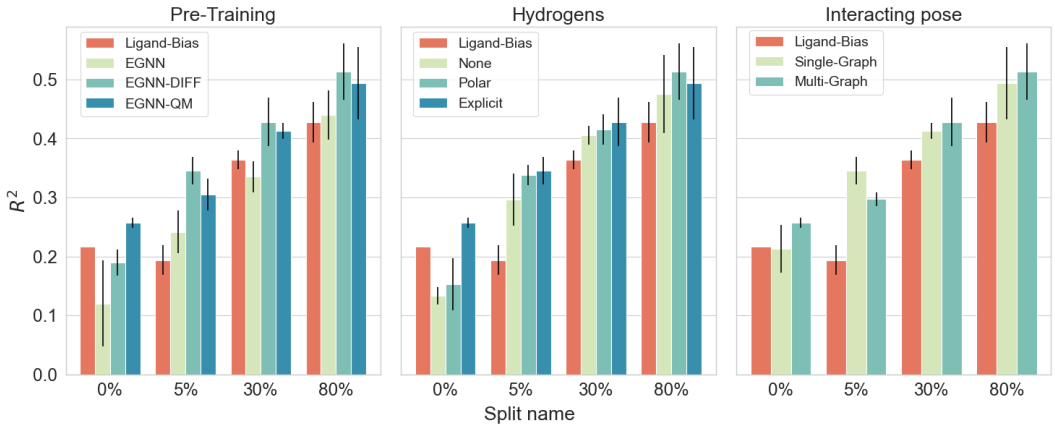


Figure 4: Effect of the EGNN additions proposed in this study on model performance. The overall performance across all eight proteins in the Case-Study-Proteins set is reported. The error bars denote the standard deviation across the three test folds. In the 0% split case, there is a only a single test fold. Due to the non-deterministic nature of training, variation in performance is due to training the same EGNN model three times. **Pre-training**: Quantum mechanical pre-training provides the greatest advantage, followed closely by diffusion pre-training. **Hydrogens**: Including explicit hydrogens is very important at low data levels. **Interacting pose**: No consistent pattern when comparing single-graph versus multi-graph.

As can be seen in Figure 4, hydrogens are very import at low data levels for generalisation. With more data, their effect becomes negligible. This is very important to keep in-mind for building models that generalise to new proteins and novel ligand chemical space.

3.2.3 Single-Graph vs. Multi-Graph

Finally, we look at the effect of the interacting pose on performance. In the default architecture (single-graph), the ligand and pocket are given to the model as a single 3D graph. In the multi-graph architecture, the ligand and the protein are first encoded with separate graphs. In theory, we expect the single-graph to outperform the multi-graph architecture.

As we can see in Figure 4, using the pose information (single-graph model) does not necessarily improve the model performance. This could either be due to the model being able to infer the interactions without the exact pose or not properly learning the interactions in the first place (and so we do not notice its effect). Either way, since these models outperform the Ligand-Bias model, they must be using the protein information in some way. Further investigation is required to understand what the models are learning.

4 Discussion

In this paper, we demonstrate that for binding affinity prediction on new proteins and chemical spaces, global 3D models outperform local models on the PDBBind dataset. Furthermore, we show that explicit hydrogen atoms in the structures and novel pre-training strategies using quantum mechanical data and diffusion modelling provide performance improvements in low data regimes for GNNs.

There are limitations to the above benchmark. This work focuses on the PDBBind dataset, a dataset comprising only crystal structures. We explicitly chose to do so to eliminate any sources of noise or error from computationally generated structures and poses. However, this has two drawbacks. First, the structures and chemical space of ligands in the dataset are not representative of the distribution of ligands in a real-world drug discovery projects. The relatively good performance of the Ligand-Bias model indicates the lack of diversity in the ligand distribution. Second, since crystal structures can only be obtained for binding ligands, this benchmark does not probe the performance of models for non-binding ligands. This is important for virtual high throughput screens where many ligands might not be binders.

In light of these limitations, it is crucial to replicate a similar analysis on additional datasets, ideally resembling real-world drug discovery datasets. If insights from this study are confirmed with more datasets, this research could represent a significant stride towards developing more universally applicable 3D binding affinity models that leverage pre-training strategies.

Acknowledgements

This work benefited greatly from the input and feedback of many colleagues. We are grateful to Ben Butt, Gail Bartlett, Richard Bradshaw, Douglas Pires, David Errington, Daniel Cutting, Emil Nichita, Jonathan Harrison, Constantin Schneider, Andrew Wedlake, Jody Barbeau for useful discussions.

References

Pedro J Ballester and John BO Mitchell. A machine learning approach to predicting protein–ligand binding affinity with applications to molecular docking. *Bioinformatics*, 26(9):1169–1175, 2010.

Ilyes Batatia, Dávid Péter Kovács, Gregor N. C. Simm, Christoph Ortner, and Gábor Csányi. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields, 2023. URL https://arxiv.org/abs/2206.07697.

Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P. Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E. Smidt, and Boris Kozinsky. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature Communications*, 13(1), May 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-29939-5. URL http://dx.doi.org/10.1038/s41467-022-29939-5.

Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

Martin Buttenschoen, Garrett M. Morris, and Charlotte M. Deane. Posebusters: Ai-based docking methods fail to generate physically valid poses or generalise to novel sequences, 2023. URL https://arxiv.org/abs/2308.05777.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

C Cortes. Support-vector networks. *Machine Learning*, 1995.

Jianyuan Deng, Zhibo Yang, Hehe Wang, Iwao Ojima, Dimitris Samaras, and Fusheng Wang. A systematic study of key elements underlying molecular property prediction. *Nature Communications*, 14(1):6395, 2023.

Guy Durant, Fergus Boyles, Kristian Birchall, Brian Marsden, and Charlotte Deane. Robustly interrogating machine learning based scoring functions: what are they learning? *bioRxiv*, pages 2023–10, 2023.

Jerome Eberhardt, Diogo Santos-Martins, Andreas F. Tillack, and Stefano Forli. Autodock vina 1.2.0: New docking methods, expanded force field, and python bindings. *Journal of Chemical Information and Modeling*, 61(8):3891–3898, July 2021. ISSN 1549-960X. doi: 10.1021/acs.jcim.1c00203. URL http://dx.doi.org/10.1021/acs.jcim.1c00203.

Exscientia. Molflux, Jan 2024a. URL https://github.com/Exscientia/molflux.

Exscientia. Physicsml, Jan 2024b. URL https://github.com/Exscientia/physicsml.

Colin R Groom, Ian J Bruno, Matthew P Lightfoot, and Suzanna C Ward. The cambridge structural database. *Structural Science*, 72(2):171–179, 2016.

Emiel Hoogeboom, Víctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3D. 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 8867–8887. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/hoogeboom22a.html.

Andrew L Hopkins, György M Keserü, Paul D Leeson, David C Rees, and Charles H Reynolds. The role of ligand efficiency metrics in drug discovery. *Nature reviews Drug discovery*, 13(2):105–121, 2014.

Dejun Jiang, Zhenxing Wu, Chang-Yu Hsieh, Guangyong Chen, Ben Liao, Zhe Wang, Chao Shen, Dongsheng Cao, Jian Wu, and Tingjun Hou. Could graph neural networks learn better molecular representation for drug discovery? a comparison study of descriptor-based and graph-based models. *Journal of cheminformatics*, 13:1–23, 2021.

Gareth Jones, Peter Willett, Robert C Glen, Andrew R Leach, and Robin Taylor. Development and validation of a genetic algorithm for flexible docking 1 1edited by f. e. cohen. *Journal of Molecular Biology*, 267(3):727–748, April 1997. ISSN 0022-2836. doi: 10.1006/jmbi.1996.0897. URL `http://dx.doi.org/10.1006/jmbi.1996.0897`.

Visvaldas Kairys, Lina Baranauskiene, Migle Kazlauskiene, Daumantas Matulis, and Egidijus Kazlauskas. Binding affinity in drug design: experimental and computational techniques. *Expert opinion on drug discovery*, 14(8):755–768, 2019.

Jie Li, Xingyi Guan, Oufan Zhang, Kunyang Sun, Yingze Wang, Dorian Bagni, and Teresa Head-Gordon. Leak proof pdbbind: A reorganized dataset of protein-ligand complexes for more generalizable binding affinity prediction, 2024. URL `https://arxiv.org/abs/2308.09639`.

Shuangli Li, Jingbo Zhou, Tong Xu, Liang Huang, Fan Wang, Haoyi Xiong, Weili Huang, Dejing Dou, and Hui Xiong. Structure-aware interactive graph neural networks for the prediction of protein-ligand binding affinity. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pages 975–985, 2021.

Zhihai Liu, Yan Li, Li Han, Jie Li, Jie Liu, Zhixiong Zhao, Wei Nie, Yuchen Liu, and Renxiao Wang. Pdb-wide collection of binding data: current status of the pdbbind database. *Bioinformatics*, 31(3): 405–412, 2015.

Yu-Chen Lo, Stefano E Rensi, Wen Torng, and Russ B Altman. Machine learning in chemoinformatics and drug discovery. *Drug discovery today*, 23(8):1538–1546, 2018.

Rocco Meli, Garrett M Morris, and Philip C Biggin. Scoring functions for protein-ligand binding affinity prediction using structure-based deep learning: a review. *Frontiers in bioinformatics*, 2: 885983, 2022.

Albert Musaelian, Simon Batzner, Anders Johansson, Lixin Sun, Cameron J. Owen, Mordechai Kornbluth, and Boris Kozinsky. Learning local equivariant representations for large-scale atomistic dynamics, 2022. URL `https://arxiv.org/abs/2204.05249`.

Tjelvar SG Olsson, Mark A Williams, William R Pitt, and John E Ladbury. The thermodynamics of protein–ligand interaction and solvation: insights for ligand design. *Journal of molecular biology*, 384(4):1002–1017, 2008.

Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features. *Advances in neural information processing systems*, 31, 2018.

Victor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) equivariant graph neural networks, 2022. URL `https://arxiv.org/abs/2102.09844`.

Michel van Kempen, Stephanie S Kim, Charlotte Tumescheit, Milot Mirdita, Jeongjae Lee, Cameron L M Gilchrist, Johannes Söding, and Martin Steinegger. Fast and accurate protein structure search with foldseek. *Nat. Biotechnol.*, 42(2):243–246, February 2024.

Mikhail Volkov, Joseph-André Turk, Nicolas Drizard, Nicolas Martin, Brice Hoffmann, Yann Gaston-Mathé, and Didier Rognan. On the frustration to predict binding affinities from protein–ligand structures with deep neural networks. *Journal of medicinal chemistry*, 65(11):7946–7958, 2022.

Zechen Wang, Liangzhen Zheng, Yang Liu, Yuanyuan Qu, Yong-Qiang Li, Mingwen Zhao, Yuguang Mu, and Weifeng Li. Onionnet-2: A convolutional neural network model for predicting protein-ligand binding affinity based on residue-atom contacting shells. *Frontiers in Chemistry*, 9, October 2021. ISSN 2296-2646. doi: 10.3389/fchem.2021.753002. URL http://dx.doi.org/10.3389/fchem.2021.753002.

wwPDB consortium. Protein data bank: the single global archive for 3d macromolecular structure data. *Nucleic acids research*, 47(D1):D520–D528, 2019.

Shuke Zhang, Yanzhao Jin, Tianmeng Liu, Qi Wang, Zhaohui Zhang, Shuliang Zhao, and Bo Shan. Ss-gnn: A simple-structured graph neural network for affinity prediction. *ACS Omega*, 8(25): 22496–22507, June 2023. ISSN 2470-1343. doi: 10.1021/acsomega.3c00085. URL http://dx.doi.org/10.1021/acsomega.3c00085.

Supplemental Information

A Single-Protein hyperparameters

We used classical architectures and features for the Single-Protein models. All the models and features below can be found in the python package MOLFLUX (Exscientia [2024a]).

Table 4: Single-Protein models

Model Name	**hyperparameters**
Random Forest	`n_estimators` $= 500$
XGBoost	`learning_rate` $= 0.2$ `subsample` $= 1$ `max_depth` $= 6$
CatBoost	`random_state` $= 0$
Support Vector Regressor	`kernel` $=$ `rbf`

Table 5: Single-Protein features

Feature name	**Comments**
Molecular Weight	
Molecular descriptors (MD)	xLogP, aromatic ring count molecular weight, num acceptors num donors, rotatable bonds, tpsa All descriptors were normalised by the train set mean and variance
ECFP	circular 2048 fingerprints
FCFP	
Topological torsion	
ECFP + MD	
FCFP + MD	
Topological torsion + MD	

B EGNN models

We used the following hyperparameters for the EGNN models.

Table 6: Single-Protein models

Hyperparameter	Value
num_layers	5
c_hidden	128
num_rbf	8
pool_type	sum
Activation	SiLU
Loss	MSELoss
Optimizer	AdamW, LR$= 5 \times 10^{-4}$
Scheduler	ReduceLROnPlateau

C Pre-training EGNN models

C.1 Quantum mechanical energy

The EGNN-QM model was pre-trained on the ccsd(t) energy of the ANI1ccx dataset. This is a dataset of roughly 500k small molecules. As is common in the domain of neural network potentials, the model was trained to predict the interaction energy (the total energy minus the self atomic energies of the atoms). The final root mean squared error of the model was 4kcal/mol.

C.2 Small molecule diffusion

The EGNN-DIFF model was trained to generate small molecules from the QM9 dataset (small molecule with up to 9 heavy atoms). This was carried out as described in the original work by Hoogeboom et al. [2022].

For use in a predictive setting, the coordinate updates in the diffusion model were turned off and a predictive pooling head was added.

C.3 Transfer learning

We match the backbones of the pre-trained models to freshly initialized models (the pooling heads are not matched, they remain randomly initialized). The backbone is then trained until convergence using early stopping at a learning rate or 5×10^{-4}. The backbone is then unfrozen and the entire model is fine-tuned at a lower learning rate 1×10^{-4}.

C.4 Training resources

Each EGNN model training takes $\sim$6 hours on a single A10 GPU.

D Metrics

Below are the explicit equations for the metrics used in the benchmarks

$$\text{Pearson Correlation Coefficient} = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2} \sqrt{\sum_j (y_j - \bar{y})^2}}$$

$$R^2 = 1 - \frac{\sum_i (x_i - y_i)^2}{\sum_j (y_j - \bar{y})^2}$$

E Additional plots

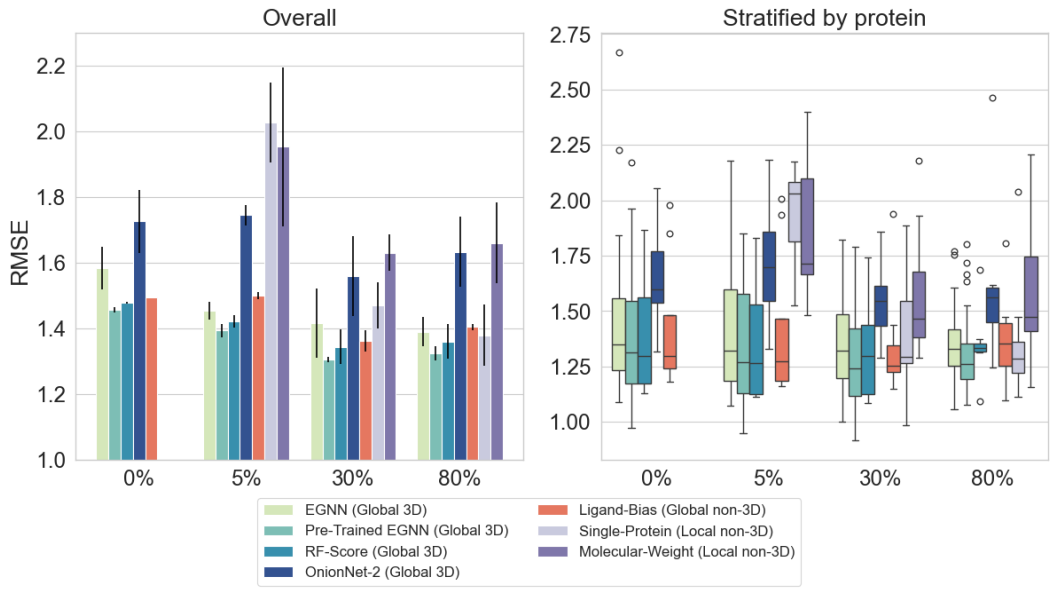


Figure 5: Overall and stratified performance at increasing train data fraction for different model families. In the low data regime, global 3D models outperform local models. Left: The error bars denote the standard deviation across the 3 test folds. Right: The boxplots represent the performance distribution over the eight proteins in the Case-Study-Proteins set.

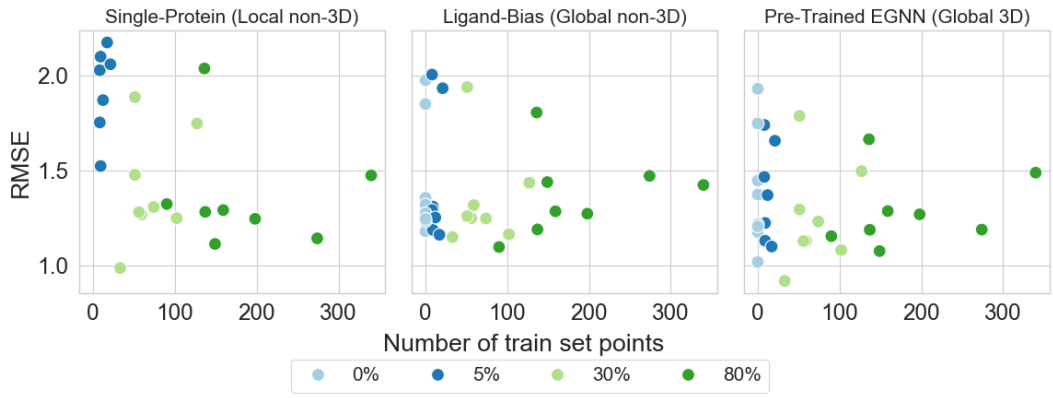


Figure 6: Effect of number of training data points on performance. Each point represents a protein from the 8 case-study proteins. The global models show a clear advantage at low data regimes.

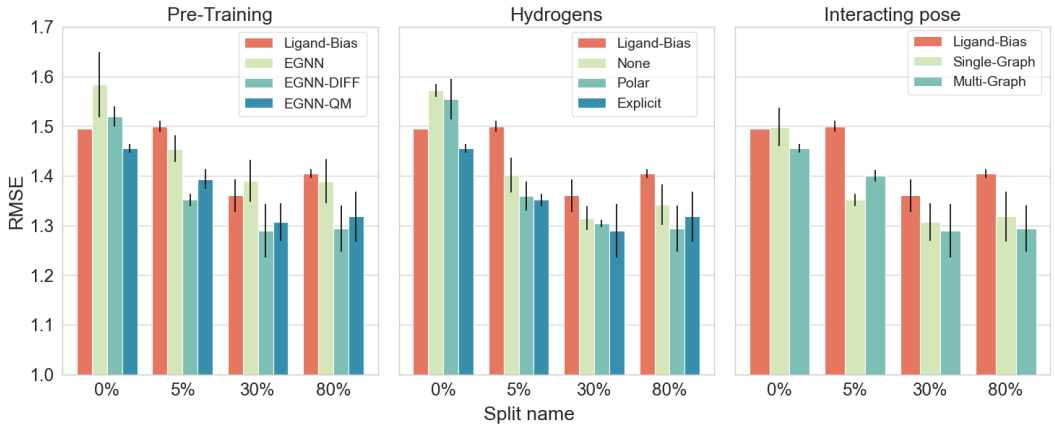


Figure 7: Effect of the EGNN additions proposed in this study on model performance. The overall performance across all eight proteins in the Case-Study-Proteins set is reported. The error bars denote the standard deviation across the 3 test folds. **Pre-training**: Quantum mechanical pre-training provides the greatest advantage, followed closely by diffusion pre-training. **Hydrogens**: Including explicit hydrogens is very important at low data levels. **Interacting pose**: No consistent pattern becomes apparent when comparing single-graph versus multi-graph.

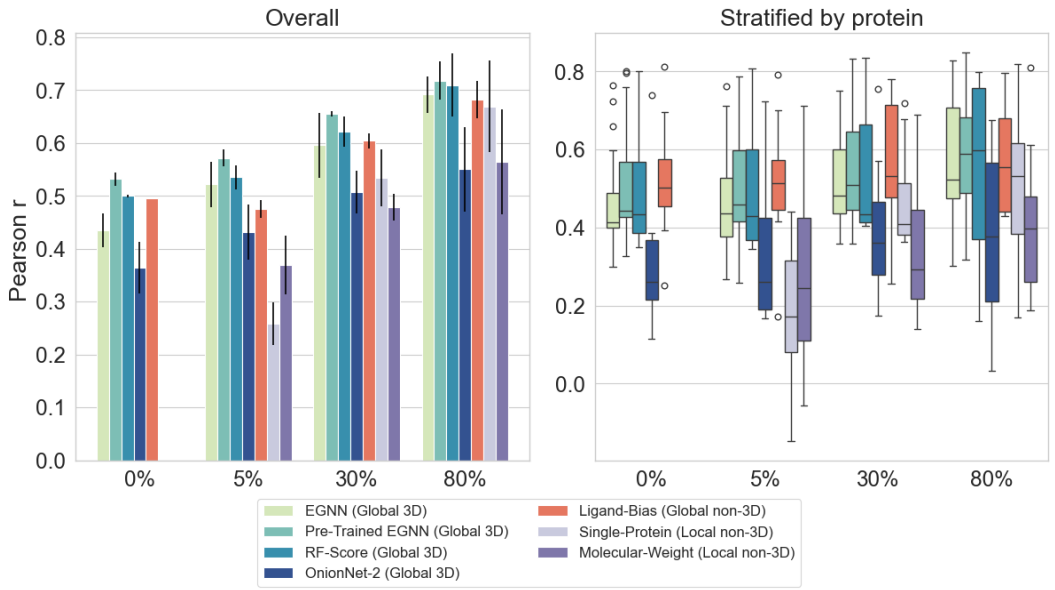


Figure 8: Overall and stratified performance at increasing train data fraction for different model families. In the low data regime, global 3D models outperform local models. Left: The error bars denote the standard deviation across the 3 test folds. Right: The boxplots represent the performance distribution over the eight proteins in the Case-Study-Proteins set.

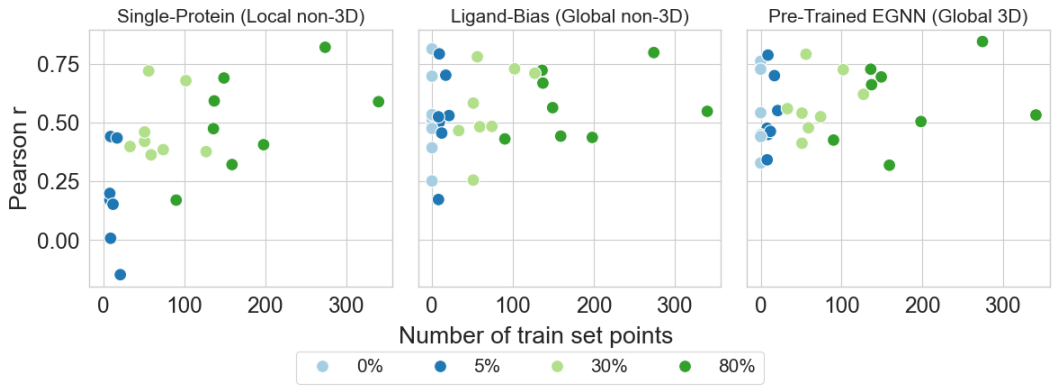


Figure 9: Effect of number of training data points on performance. Each point represents a protein from the 8 case-study proteins. The global models show a clear advantage at low data regimes.

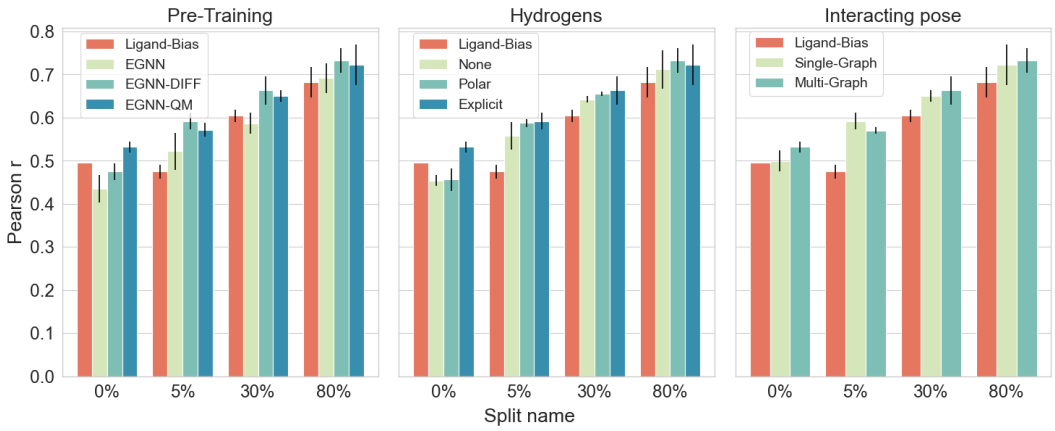


Figure 10: Effect of the EGNN additions proposed in this study on model performance. The overall performance across all eight proteins in the Case-Study-Proteins set is reported. The error bars denote the standard deviation across the 3 test folds. **Pre-training**: Quantum mechanical pre-training provides the greatest advantage, followed closely by diffusion pre-training. **Hydrogens**: Including explicit hydrogens is very important at low data levels. **Interacting pose**: No consistent pattern becomes apparent when comparing single-graph versus multi-graph.