Impacts of Data and Models on Unsupervised Pre-training for Molecular Property Prediction

Elizabeth Coda [1,2] **Gihan Panapitiya**[1] **Emily Saldanha**[1]
[1]Pacific Northwest National Laboratory [2]University of California, San Diego
{first.last}@pnnl.gov

Abstract

The available labeled data to support molecular property prediction are limited in size due to experimental time and cost requirements. However, unsupervised learning techniques can leverage vast databases of molecular structures, thus significantly expanding the scope of training data. We compare the effectiveness of pre-training data and modeling choices to support the downstream task of molecular aqueous solubility prediction. We also compare the global and local structure of the learned latent spaces to probe the properties of effective pre-training approaches. We find that the pre-training modeling choices affect predictive performance and the latent space structure much more than the data choices.

1 Introduction

The unsupervised and self-supervised pre-training of large-scale foundation models has led to dramatic progress across a range of fields, especially in domains relying on natural language processing Devlin et al. (2019); Brown et al. (2020) and computer vision Radford et al. (2021). Such training approaches have a natural application in molecular property prediction due to the challenges of collecting large datasets of measured target properties and the availability of massive unlabeled molecular structure datasets Kim et al. (2018); Irwin & Shoichet (2005). Moreover, AI-guided design stands to benefit from improved predictive models as errors in property prediction will propagate to the use of these models for molecular design. There have been many recent efforts to develop effective pre-training approaches for molecular structure Morris et al. (2020); Gómez-Bombarelli et al. (2018); Colby et al. (2019); Jin et al. (2020); Chithrananda et al. (2020); Rong et al. (2020). We aim to study the impacts of data and modeling choices on the effectiveness of pre-training strategies for the task of molecular aqueous solubility prediction. We analyze the impact of data properties including size, diversity, and similarity to the target property data across a range of recent pre-training methods for learning from molecular structure including SMILES-based versus graph-based methods trained on masking, autoencoder, and translation objectives. We find that pre-training methods are a much stronger driver of the ultimate performance than data choices, with the best performing pre-training approaches being relatively insensitive to the choice of pre-training dataset.

2 Data

We leverage the solubility dataset from Panapitiya et al. (2022), which consists of the measured aqueous solubility of $\sim$17k molecules, in units of mol/L. We aim to predict log S, the base 10 logarithm value of solubility. The log S values range from -17.46 mol/L to 1.66 mol/L with a median value of -2.74 mol/L and a standard deviation of 2.24 mol/L.

To probe the important pretraining dataset properties, we develop four pre-training molecular training datasets from the PubChem Compound database Kim et al. (2018) that vary along in size, diversity, and similarity to the target property data. The similar-small (SimS) and similar-large (SimL) datasets

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

Table 1: Summary of the size, diversity, and similarity metrics for different pre-training datasets.

Dataset	Size	Diversity	Distance from Solubility
Solubility	17k	5.82	0.00
SimS	400k	6.92	1.76
SimL	1M	6.98	2.17
OOD	1M	7.20	9.56
Rand	1M	7.07	7.85
USPTO	479k	6.90	4.90
QMugs	665k	6.86	13.57

were generated by sampling PubChem molecules that are similar to the molecules contained within our target solubility dataset, as measured by the fingerprint similarity function in RDKit RDKit, online. The OOD dataset was generated by sampling PubChem molecules that are out of the distribution (OOD) relative to the target solubility dataset. We have used the cosine similarity between molecular embeddings obtained from a trained model to determine which molecules are OOD rather than the fingerprint similarity because experimentally, this method sampled a wider range of molecular structures. Lastly, the random (Rand) dataset is a random sample of PubChem molecules. Full details of the sampling procedures used to construct each of these datasets are available in Appendix A.1.

In addition to pre-training on these constructed datasets, we utilize publicly available pre-trained models trained on the USPTO dataset Wang et al. (2021), which contains data from chemical reactions, and the QMugs (Quantum-Mechanical Properties of Druglike Molecules) dataset Isert et al. (2022), which contains 3D molecular geometry data.

Table 1 contains metrics summarizing the diversity of each dataset and its similarity to the target property dataset. To quantify the internal diversity of each pre-training dataset, we use the entropy estimator described in Leguy et al. (2021). To quantify the similarity between each pre-training dataset and the target solubility dataset, we calculate the Fréchet ChemNet Distance (FCD) Brown et al. (2019).

3 Pre-training Modeling Approaches

To compare the effectiveness of both data and modeling choices, we pre-train the models described below on our constructed molecular datasets. From each of these pre-trained models, we can extract a latent embedding of any input molecule. We use these representations in fine-tuning and in our analysis. Further details are available in Appendix A.2.

We utilize five general pre-training approaches: variational autoencoders (VAEs), representational translation, structural masking, reaction aware learning, and 3D learning. VAEs are tasked to reconstruct the molecular structure based on a learned compressed latent representation of the molecule. Specifically, we leverage two different VAE architectures - a RNN and a CNN Gómez-Bombarelli et al. (2018); Colby et al. (2019). We also consider a Hierarchical VAE (HVAE) which takes in a graph representation of each molecule at multiple resolutions Jin et al. (2020). However, the computational cost of this model limits our use of it. Representational translation utilizes a transformer model with the pre-training task of translating SMILES strings to IUPAC strings Morris et al. (2020). With structural masking models, certain segments of inputs are masked and the pretraining task is to predict these masked segments. We use ChemBERTa Chithrananda et al. (2020), which uses tokenized SMILES strings as inputs, and GROVER Rong et al. (2020) which uses molecular graphs as inputs. Reaction-aware learning as implemented in the MolR model Wang et al. (2021) is a GNN model where the pretraining task is to train a model that preserves the sum of molecular embeddings of the product molecules and the reactant molecules. Finally, the 3DInfomax model Stärk et al. (2021) compares embeddings of a model whose input is 3D molecules and a GNN, whose input is the 2D graph representation of molecules. Due to data availability and computational constraints, we do not pretrain MolR or 3DInfomax on all our datasets but use the publicly available pretrained models.

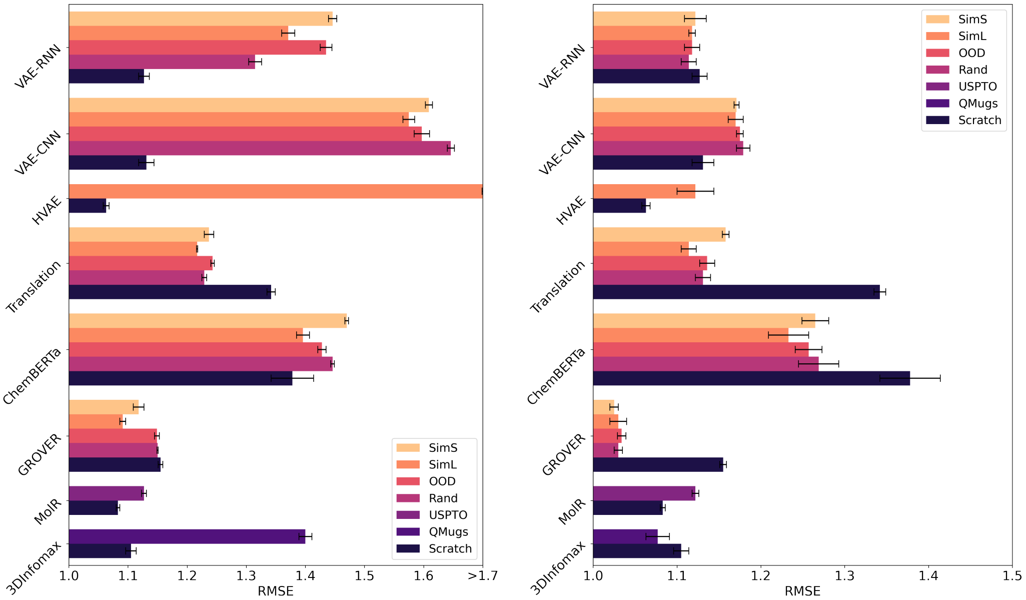

Figure 1: (Left) Finetuning the predictor only. (Right) Finetuning the full model. Results are also available in Table 6

4 Fine-tuning Results

After pre-training, we add a two-layer feed-forward neural network to each model. We have two fine-tuning methods for solubility prediction. In (FT - Pred Only), we freeze the entire pre-trained model and only train the predictor network. The learned latent embeddings are frozen in this method, which allows us to explore how much chemical knowledge is encoded into the embedding using the pre-training objectives alone. In (FT - Full Model), we allow the weights in the pre-trained model to update along with the predictor network, which allows the model to extract new chemical knowledge from the molecular structure based on the supervised objective. We report the average root mean squared error (RMSE) ± its standard deviation over five runs in Figure 1 and in Table 6. To separate the benefits of pre-training from the model versus the strengths of the individual model architectures, we also train each model from scratch (FS) on the target solubility dataset.

4.1 Fine-tuning Performance

As seen in Table 6, fine-tuning the entire model and predictor network always outperformed fine-tuning only the predictor network. This suggests that after pre-training, the learned embeddings do not have enough information to predict solubility and must adapt the learned representations during fine-tuning to incorporate the necessary information. Overall, GROVER was the best model with a RMSE ranging from 1.02 to 1.03 log S, depending on the pre-training dataset, about 0.12 lower compared to GROVER trained from scratch. HVAE trained from scratch was also competitive, with a RMSE of 1.06. Both GROVER and HVAE rely on a graph representation of the molecule in contrast to the SMILES representation used by the other models, which all had an RMSE of at least 1.11, regardless of the pre-training dataset. Pre-training was ultimately not helpful for either of the VAE models. For the VAE-RNN the effect of pre-training was negligible while for the VAE-CNN pre-training was surprisingly harmful to the fine-tuned performance. Pre-training was also harmful with the HVAE. Compared to training from scratch, pre-training significantly helped the remaining transformer based models, Translation and ChemBERTa. However, the fine-tuned RMSE for these models was much larger than the RMSE of the best models.

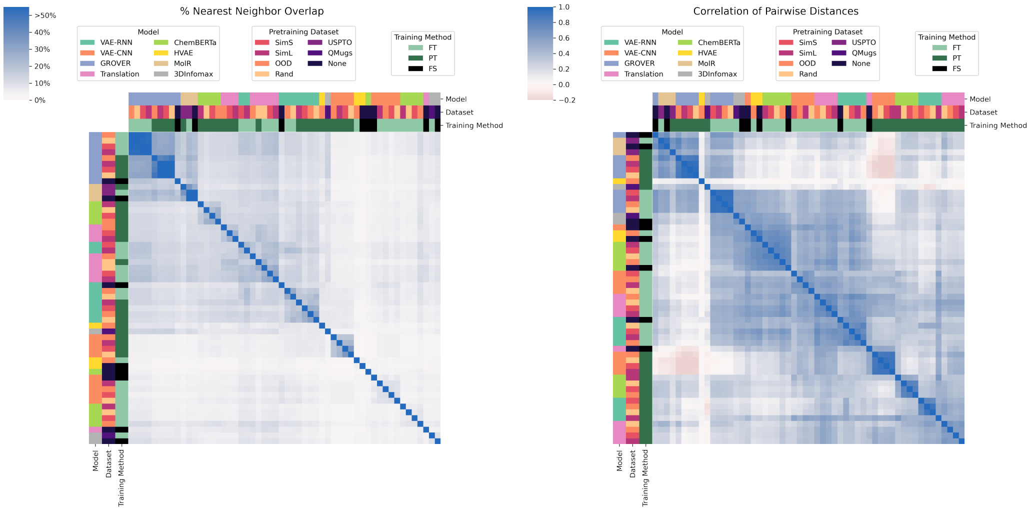

Figure 2: (Left) Percentage of 10-nearest neighbors overlap, averaged over all molecules. (Right) Pairwise distance Spearman correlations. FT = fine-tuned, PT = pre-trained, FS = from scratch.

4.2 Impact of Data Properties

To analyze the impact of pre-training dataset size on downstream performance, we perform a pairwise comparison of the models pre-trained on the SimS and SimL datasets, which have comparable diversity and similarity metrics as shown in Table 1. When we fine-tuned only the predictor network, the latent embeddings trained on SimL outperformed the embeddings trained on SimS on all models. However, the improvement on SimL was small on the better performing models, GROVER and Translation. When we fine-tuned the full model, pre-training on a larger dataset helped only the SMILES-based transformer models, Translation and ChemBERTa. While pre-training was not helpful for the VAE models and we therefore do not expect to observe much sensitivity to the pre-training dataset, GROVER was also insensitive to the pre-training dataset.

We use the SimL (more similar) and Rand (less similar) datasets to compare the impact of pre-training on a dataset that is similar to the target solubility dataset holding dataset size and dataset diversity approximately constant. When we fine-tuned the predictor only, on all but the VAE-RNN, the embeddings trained on SimL outperform the embeddings trained on Rand. These effects are diminished when we fine-tune the entire model, where pre-training on a similar dataset helped only the SMILES-based transformer models.

Finally, we look at the OOD (more diverse) and Rand (less diverse) datasets to measure the role of pre-training dataset diversity. While these datasets do have different similarity scores, they are the best dataset pairing we could find that have different diversity scores and similarity scores that are close together. We find that all models are generally unaffected by the differences in these two pre-training datasets with both fine-tuning methods.

5 Latent Space Analysis

Using the learned molecular embeddings, we analyze the differences in information encoded in the latent space that results from different modeling approaches and pre-training datasets to understand how latent space properties are related to fine-tuning performance. For this analysis, we use the embeddings of the molecules from the target solubility dataset.

To compare the global structures of two latent spaces, we look at the Spearman correlation of their pairwise molecular cosine similarities. The results are summarized in the right plot of Figure 2. We find a hierarchical clustering behavior, with the latent spaces first clustering by training method and then by modeling approach. However, compared to the rest of the models, the GROVER model learns a different global structure and shows the unique behavior of having high alignment of its pre-trained and fine-tuned spaces, indicating the initially learned embedding is not strongly adjusted during the fine-tuning process.

To compare the local structures of two latent spaces, we compare the nearest neighbors of a given molecule in each latent space. Specifically, let A_i be the set of the k nearest neighbors of molecule i in one latent space, and let B_i be the set of k nearest neighbors of molecule i in another latent space. To obtain a single metric for the latent space pairing, we find the percentage of nearest neighbors that overlap averaged over all molecules, $\sum_i \frac{|A_i \cap B_i|}{|A_i \cup B_i|}$.

Using $k = 10$ and embedding cosine similarity as the distance metric, the left plot of Figure 2 shows the results. Similar to the global structure, we find that the local structure of the GROVER latent spaces is also relatively stable between the pre-trained and fine-tuned versions. Consistent with the predictive performance, we find that local structure is relatively stable to different pre-training datasets but is strongly affected by different modeling choices.

6 Conclusions

We find that pre-training methods are a much stronger driver of pre-training effectiveness for solubility prediction than pre-training dataset choices. Of the models explored, the string-based transformers are most sensitive to data properties. We also find that the best performing model, GROVER, performs the least adjustment of its latent space during fine-tuning, indicating that it has successfully encoded most of the relevant chemical knowledge into the local and global structure of the latent space during pre-training.

One limitation of this work is that the failure to identify strong impacts of data properties may be because the range of data properties that we explored was too limited and that significantly larger or more diverse data would be needed to see a strong impact on performance. Additionally, we have demonstrated the relative performance of these data and modeling choices for only an individual target property. Future work to validate these patterns across a broader array of target properties will be needed.

References

Brown, N., Fiscato, M., Segler, M. H., and Vaucher, A. C. Guacamol: Benchmarking models for de novo molecular design. *Journal of Chemical Information and Modeling*, 59(3):1096–1108, 2019. doi: 10.1021/acs.jcim.8b00839. URL `https://doi.org/10.1021/acs.jcim.8b00839`. PMID: 30887799.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf`.

Chithrananda, S., Grand, G., and Ramsundar, B. Chemberta: Large-scale self-supervised pretraining for molecular property prediction, 2020.

Colby, S. M., Nuñez, J. R., Hodas, N. O., Corley, C. D., and Renslow, R. R. Deep learning to generate in silico chemical property libraries and candidate molecules for small molecule identification in complex samples, 2019.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL `https://aclanthology.org/N19-1423`.

Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 02 2018. doi: 10.1021/acscentsci.7b00572. URL `https://doi.org/10.1021/acscentsci.7b00572`.

Irwin, J. J. and Shoichet, B. K. Zinc- a free database of commercially available compounds for virtual screening. *Journal of chemical information and modeling*, 45(1):177–182, 2005.

Isert, C., Atz, K., Jiménez-Luna, J., and Schneider, G. Qmugs, quantum mechanical properties of drug-like molecules. *Scientific Data*, 9(1), 2022. doi: 10.1038/s41597-022-01390-7.

Jin, W., Barzilay, R., and Jaakkola, T. Hierarchical generation of molecular graphs using structural motifs, 2020.

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 2019 update: improved access to chemical data. *Nucleic Acids Research*, 47(D1):D1102–D1109, 10 2018. ISSN 0305-1048. doi: 10.1093/nar/gky1033. URL `https://doi.org/10.1093/nar/gky1033`.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. 2015. URL `http://arxiv.org/abs/1412.6980`.

Leguy, J., Glavatskikh, M., Cauchy, T., and Da Mota, B. Scalable estimator of the diversity for de novo molecular generation resulting in a more robust qm dataset (od9) and a more efficient molecular optimization. *Journal of Cheminformatics*, 13(1):76, 2021. doi: 10.1186/s13321-021-00554-8. URL `https://doi.org/10.1186/s13321-021-00554-8`.

Morris, P., St. Clair, R., Hahn, W. E., and Barenholtz, E. Predicting binding from screening assays with transformer network embeddings. *Journal of Chemical Information and Modeling*, Jun 2020. ISSN 1549-9596. doi: 10.1021/acs.jcim.9b01212. URL `https://doi.org/10.1021/acs.jcim.9b01212`.

Panapitiya, G., Girard, M., Hollas, A., Sepulveda, J., Murugesan, V., Wang, W., and Saldanha, E. Evaluation of deep learning architectures for aqueous solubility prediction. *ACS Omega*, April 2022. doi: 10.1021/acsomega.2c00642. URL `https://doi.org/10.1021/acsomega.2c00642`.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

RDKit, online. RDKit: Open-source cheminformatics. `http://www.rdkit.org`.

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

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

Wang, H., Li, W., Jin, X., Cho, K., Ji, H., Han, J., and Burke, M. D. Chemical-reaction-aware molecule representation learning, 2021. URL `https://arxiv.org/abs/2109.09888`.

A Appendix

A.1 Datasets

Here we provide a more detailed outline of the procedures used to generate each of the datasets from the PubChem database (310 files with 500k molecules each) Kim et al. (2018).

SimS and SimL For each file in the PubChem database, we randomly sample 60k molecules. We additionally sample 3k molecules from our property database. For every molecule in our property sample, we find its nearest neighbor (NN) in the 60k subset and add it to the SimS dataset. The molecule's three NNs are added to the SimL dataset. Distance is measured using RDKit's fingerprint similarity. Because of repeat NNs, the SimS dataset contains approximately 400k molecules and the SimL dataset contains approximately 1M molecules.

OOD For each file in the PubChem database, we randomly sample a 60k subset of molecules. We find the similarity score between each sampled PubChem molecule and the target solubility database and we add the 3225 PubChem molecules with the smallest similarity scores to the OOD dataset. We define the similarity score between a PubChem molecule m and the target solubility database D as $max\{s(z_m, z_x)|x \in D\}$ where s is the cosine similarity and z_m is the latent embedding of a molecule obtained using the GROVER model pre-trained on the SimL dataset. Repeating this over all files results in a dataset of approximately 1M molecules.

Rand The random (Rand) dataset is a 1M molecule random sample from the PubChem-10M pre-training dataset used to pre-train the ChemBERTa model Chithrananda et al. (2020).

A.2 Pre-training Models

A summary of the models and the time required for pre-training is summarized in Table 2. All training was done on a single GPU except for VAE-CNN and Translation, which trained on two GPUs. We have attempted follow the pre-training procedure and hyperparameters used in the original implementations of these methods as closely as possible, but list the full hyperparameters we used in Table 3.

Table 2: Overview of the pre-training models.

Method	Architecture	Data Representation	Parameters	Time to Pre-train	Latent Dimension
VAE-RNN	RNN	SMILES	4.6M	$<$ 1 day	32
VAE-CNN	CNN	SMILES	8.8M	$<$ 1 day	128
HVAE	GNN	Graph	20M	$\sim$ 1 month	32
Translation	Transformer	SMILES	44M	$\sim$ 1 day	512
ChemBERTa	Transformer	SMILES	83M	$\sim$ 1 day	768
GROVER	Transformer	Graph	2.2M	$\sim$ 2 weeks	370
MoLR	GNN	Graph	2.2M	NA	1024
3DInfomax	GNN	Graph	5M	NA	256

Table 3: Summary of the pre-training hyperparameters.

Method	Epochs	Batch Size	Initial Learning Rate	Weight Decay
VAE-RNN	50	256	1e-3	-
VAE-CNN	500	256	1e-3	1e-6
HVAE	20	8	1e-3	-
Translation	100	24	1e-4	-
ChemBERTa	10	8	5e-5	-
GROVER	350	512	1.5e-4	1e-7

A.3 Fine-tuning

Fine-tuning hyperparameters for both fine-tuning methods are listed in Tables 4 and 5. Learning rate and weight decay for the Adam optimizer Kingma & Ba (2015) were determined using random grid search. For fine-tuning on the predictor only, we fine-tune with a batch size of 256 for 5k epochs with early stopping. For fine-tuning on the full model, we fine-tune with the batch size and number of epochs indicated in the table, with early stopping. We repeat fine-tuning five times on each model to capture a sense of the variability of the model.

For fine-tuning on the full model, we included the pre-training task loss in the fine-tuning loss for some models. These models that were jointly trained on both the pre-training and solubility prediction task are identified by the 'Joint' column in Table 5. We used joint training on models it was helpful for and otherwise did not use joint training.

Table 4: Summary of the fine-tuning hyperparameters used in fine-tuning the predictor only.

Method	Dataset	Learning Rate	Weight Decay
VAE-RNN	SimS	9.72e-5	4.72e-6
	SimL	7.66e-2	4.94e-7
	OOD	6.89e-4	8.04e-8
	Rand	8.99e-4	3.91e-8
VAE-CNN	SimS	9.66e-4	9.16e-8
	SimL	4.98e-4	3.94e-8
	OOD	8.61e-4	4.97e-7
	Rand	2.85e-3	5.02e-7
HVAE	SimL	1.94e-5	2.28e-7
Translation	SimS	4.04e-3	3.30e-8
	SimL	8.25e-3	8.75e-6
	OOD	7.57e-3	2.70e-8
	Rand	2.04e-3	2.17e-6
ChemBERTa	SimS	2.57e-4	8.42e-8
	SimL	7.53-4	1.57e-9
	OOD	7.88e-4	2.45e-7
	Rand	1.79e-4	3.04e-7
GROVER	SimS	3.18e-4	3.77e-9
	SimL	5.30e-4	3.99e-9
	OOD	5.98e-4	6.15e-8
	Rand	9.60e-4	4.09e-8
MolR	USPTO	6.80e-5	5.18e-8
3DInfomax	QMugs	6.40e-3	9.39e-9

Table 5: Summary of the fine-tuning hyperparameters used in fine-tuning the full model. Note that the hyperparameters for the HVAE model were inadvertently not saved and could not be reproduced due to the high computational requirements of this model.

Method	Epochs	Batch Size	Joint	Dataset	Learning Rate	Weight Decay
VAE-RNN	200	256	Yes	SimS	2.50e-3	1.00e-6
				SimL	3.00e-3	1.00e-6
				OOD	7.25e-3	8.04e-8
				Rand	4.78e-3	5.77e-6
				Scratch	2.50e-3	1.00e-6
VAE-CNN	50	256	No	SimS	2.79e-4	8.72e-7
				SimL	1.00e-3	1.00e-8
				OOD	2.73e-4	9.78e-9
				Rand	8.59e-4	5.03e-8
				Scratch	1.00e-3	1.00e-8
HVAE	50	-	No	SimL	-	-
				Scratch		
Translation	50	16	Yes	SimS	8.59e-4	5.76e-6
				SimL	2.00e-4	1.00e-6
				OOD	2.56e-4	3.78e-7
				Rand	6.52e-4	8.83e-9
				Scratch	3.88e-4	7.80e-7
ChemBERTa	10	64	No	SimS	6.71e-5	3.78e-9
				SimL	8.12e-5	5.47e-8
				OOD	8.81e-5	8.84e-6
				Rand	9.13e-5	8.96e-7
				Scratch	2.84e-5	9.66e-8
GROVER	200	32	No	SimS	2.10e-4	1.20e-7
				SimL	2.27e-5	1.96e-6
				OOD	3.74e-4	3.08e-9
				Rand	3.13e-5	4.36e-7
				Scratch	3.29e-5	9.33e-9
MolR	200	512	No	USPTO	6.67e-6	9.86e-10
				Scratch	9.60e-5	7.93e-8
3DInfomax	1000	8	No	QMugs	7.97e-4	5.23e-9
				Scratch	9.96e-4	3.86e-7

A.4 Full Results

Table 6: Summary of the transfer learning results on each model and dataset. We report the RMSE when the model is trained from scratch (FS) as well as the RMSE after pre-training on the indicated dataset. The FT-Pred Only column refers to fine-tuning (FT) only the predictor network while the FT-Full Model column refers to fine-tuning the complete model.

Method	Data	FS	FT-Pred Only	FT-Full Model
VAE - RNN	SimS	1.127 ± 0.009	1.446 ± 0.007	1.122 ± 0.013
VAE - RNN	SimL		1.371 ± 0.011	1.118 ± 0.004
VAE - RNN	OOD		1.435 ± 0.010	1.118 ± 0.009
VAE - RNN	Rand		1.315 ± 0.011	1.114 ± 0.009
VAE - CNN	SimS	1.131 ± 0.013	1.609 ± 0.006	1.171 ± 0.003
VAE - CNN	SimL		1.575 ± 0.010	1.170 ± 0.009
VAE - CNN	OOD		1.597 ± 0.013	1.175 ± 0.004
VAE - CNN	Rand		1.646 ± 0.006	1.179 ± 0.008
HVAE	SimL	**1.063 ± 0.005**	2.251 ± 0.001	1.122 ± 0.022
Translation	SimS	1.342 ± 0.007	1.237 ± 0.008	1.158 ± 0.004
Translation	SimL		1.217 ± 0.001	1.114 ± 0.009
Translation	OOD		1.243 ± 0.003	1.136 ± 0.009
Translation	Rand		1.229 ± 0.004	1.131 ± 0.009
ChemBERTa	SimS	1.378 ± 0.036	1.470 ± 0.003	1.265 ± 0.016
ChemBERTa	SimL		1.396 ± 0.011	1.233 ± 0.024
ChemBERTa	OOD		1.428 ± 0.007	1.257 ± 0.016
ChemBERTa	Rand		1.446 ± 0.003	1.269 ± 0.024
GROVER	SimS	1.155 ± 0.004	1.118 ± 0.009	**1.025 ± 0.005**
GROVER	SimL		**1.091 ± 0.005**	1.030 ± 0.010
GROVER	OOD		1.149 ± 0.004	1.034 ± 0.005
GROVER	Rand		1.150 ± 0.001	1.030 ± 0.005
MolR	USPTO	1.083 ± 0.003	1.127 ± 0.004	1.122 ± 0.004
3DInfomax	QMugs	1.105 ± 0.009	1.400 ± 0.011	1.077 ± 0.014