Efficient Cross-Functional Learning for Atomistic Modeling of Materials

Anonymous Authors[1]

Abstract

Adapting atomistic foundation models to higher-fidelity DFT functionals is limited by the scarcity and cost of target labels. In practice, models are often pretrained on large datasets computed with lower-cost functionals, such as PBE, and then fine-tuned on smaller datasets generated with higher-accuracy but more computationally expensive functionals like r2SCAN. However, there is limited understanding of how to efficiently perform such cross-functional transfer, especially how the choice of adaptation strategy interacts with target data availability and compute constraints. In this work, we compare two cross-functional adaptation strategies for atomistic foundation models under realistic data and compute constraints: (i) a simple and lightweight delta-learning method for efficient adaptation in low-data regimes, and (ii) a multi-head fine-tuning that jointly learns multiple functionals through a shared representation. Through layer-wise probing, we show that delta-learning is the most effective strategy in low-data, low-compute settings, while multi-head learning is most useful when supervision from multiple functionals is available. These results provide a practical recipe for cross-functional adaptation under different resource constraints.

1. Introduction

Atomistic foundation models can approximate DFT across large numbers of structures, making large-scale materials screening feasible (Duval et al., 2023; Rhodes et al., 2025; Deng et al., 2023). Their practical value, however, depends on adaptation. In many practical workflows, large pretraining datasets are available only for lower-cost DFT functionals such as PBE, while the downstream targets of interest are computed with higher-fidelity and more expensive functionals such as r2SCAN (Perdew et al., 1996; Sun et al., 2015; Furness et al., 2020). This creates a persistent bottleneck: high accuracy target labels are scarce, costly, and often too limited to support heavy post-training.

Cross-functional transfer is not only a small-data problem. Different functionals can induce shifted energy scales and partially mismatched supervision signals, making adaptation nontrivial even for strongly pretrained models (Huang et al., 2025). In this regime, the practical issue is to choose an adaptation strategy that uses limited supervision effectively while remaining computationally efficient (Radova et al., 2025).

We study two complementary strategies for this setting: delta-learning, which learns a lightweight correction on top of frozen pretrained representations (Ramakrishnan et al., 2015; Pitfield et al., 2025), and multi-head fine-tuning, which jointly learns multiple functionals through a shared backbone and functional-specific heads (Shoghi et al., 2023; Zhang et al., 2026). We also use layer-wise probing to identify which parts of the pretrained representation remain the most transferable across functionals, and to explain why lightweight adaptation can work. Across MatPES (Kaplan et al., 2025) and LeMatBulk (Siron et al., 2025), our results show distinct roles: delta-learning is the strongest option under tight data and compute budgets, whereas multi-head learning is more useful when supervision from multiple functionals can be exploited jointly.

The main contributions of this work are summarized as follows:

1. We study cross-functional adaptation of pretrained atomistic models in the regime where target labels and post-training compute are both limited.
2. We propose a lightweight delta-learning correction strategy that combines frozen intermediate representations with the model's own predictions, and show through ablations that both components are important.
3. We identify that delta-learning is the strongest option under tight data and compute budgets, whereas multi-head fine-tuning is effective when supervision from

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

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

multiple functionals can be exploited jointly, allowing bi-directional transfer between higher and lower-fidelity tasks.

2. Related Work

Prior work on atomistic transfer learning has shown that pretraining can reduce the amount of task-specific data needed for downstream prediction. In materials modeling, transfer from large simulation datasets to smaller target tasks has been shown to improve accuracy and sample efficiency (Hoffmann et al., 2023), while the choice of which layers or parameters to adapt can strongly affect downstream performance (Kolluru et al., 2022; Pinto, 2025). More recent benchmark studies further show that fine-tuning pretrained graph models is not straightforward in sparse-label regimes, where robustness and overfitting become central concerns (Liu et al., 2025). These observations motivate our focus on adaptation strategies that remain effective when target supervision is limited.

A second line of work studies how atomistic models benefit from joint learning across tasks, datasets, and levels of theory. Multi-task training has been shown to improve robustness both during pretraining and during downstream adaptation (Shoghi et al., 2023; Zhang et al., 2026). In the more specific setting of cross-functional transfer, Huang et al. (2025) showed that transfer from lower-fidelity to higher-fidelity DFT labels is feasible, while also highlighting the difficulties introduced by energy-scale shifts and weak correlations across functionals. More broadly, recent multi-fidelity and joint-training approaches suggest that sharing supervision across tasks can improve performance (Wood et al., 2025; Batatia et al., 2025).

Moreover, delta-learning exploits correlations between low and high-fidelity predictions by learning only a correction term. This idea was introduced in quantum chemistry by Ramakrishnan et al. (2015) and later extended to low-data and atomistic settings (Hutchinson et al., 2017; Shimakawa et al., 2024; Pitfield et al., 2025). Such methods are especially attractive when target labels or post-training compute are severely limited, since they avoid updating most of the underlying model.

Our paper compares these two distinct strategies (lightweight correction or joint learning across functionals) in a common pretrained backbone and uses layer-wise probing to understand why their behavior differs across data and compute regimes.

3. Methods

3.1. Delta-learning

Our goal is to devise a method that can make models adaptable with minimal data and compute. We would like to leverage what the foundation model has learned during training, retraining as little weights as possible. This can be done by adding a trainable layer like in probing. Ideally, we would like to use the lowest dimension input to the trained probe to make it efficient in extreme data scarcity scenarios. We therefore choose to take as input the prediction of the foundation model (a single number) directly as an input to the probe as well as some embeddings of the molecule. We select the right embedding using layer-wise probing (see section 3.2) and ablate the use of the foundation model's prediction in section 4.3.

Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ be the matrix which contains the embedding of each atom of a molecule with rows $\mathbf{x}_i \in \mathbb{R}^d$ for $i \in \{1, \dots, n\}$. We train a small model to predict the correction to apply to Orb-v3 (Rhodes et al., 2025) given the embeddings. Formally, if E_Δ is trained to satisfy :

$$E_\Delta\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i,\ E_{\mathrm{Orb}}(\mathbf{X})\right) = E_{\mathrm{DFT}}(\mathbf{X}) - E_{\mathrm{Orb}}(\mathbf{X}).$$

Here, the model takes as input the average of each atom's embeddings and the predicted energy, it outputs the correction to apply to the Orb-v3 prediction. The model can be a simple Ridge regression (referred to as Ridge regression delta-learning in the rest of this paper) or an MLP (MLP delta-learning). The goal of this method is to achieve data and compute efficiency thanks to the fact that the number of trained parameters is minimal. More details on method and hyperparameters can be found in Appendix B.2.

3.2. Layer-wise probing of frozen representations

A central question in cross-functional transfer is which representations of a pretrained model remain most robust under domain shift. While deeper layers may encode features that are more refined for the pretraining objective, they may also become increasingly specialized and less transferable. We study this with layer-wise probing of frozen Orb-v3 representations. For each structure, we extract graph-level representations from the encoder output and from each message-passing block by mean-pooling atom-level embeddings. We then train lightweight probes to predict formation energy per atom from each frozen representation.

We use probe performance as a layer-selection criterion: the layer that best preserves transferable information under frozen readout is selected for the downstream delta-learning head.

Implementation details for the probing protocol are given in

Appendix B.3.

3.3. Multi-head learning

We fine-tune a single shared Orb-v3 backbone jointly on multiple DFT functionals and attach one prediction head per functional.

Formally, given a molecular graph $\mathcal{G}$, the backbone produces node embeddings:

$$\mathbf{h}_i = \Phi_{\text{backbone}}(\mathcal{G}),$$

which are aggregated into a graph-level representation:

$$\mathbf{h}_{\mathcal{G}} = \sum_i \mathbf{h}_i.$$

Each functional $k \in \mathcal{K}$ is associated with a dedicated prediction head:

$$E^{(k)}_{\text{pred}} = \text{Head}_k(\mathbf{h}_{\mathcal{G}}).$$

Loss function. Training is performed using a weighted multi-task regression objective:

$$\mathcal{L} = \sum_{k \in \mathcal{K}} \lambda_k \mathcal{L}_k \qquad \text{with} \qquad \mathcal{L}_k = \|E^{(k)}_{\text{pred}} - E^{(k)}_{\text{DFT}}\|^2.$$

An important challenge in cross-functional learning is that different DFT functionals produce energy targets with distinct scales and offsets, which can create task imbalance and unstable optimization when training jointly. We define the multi-head objective in task-normalized target spaces to reduce imbalance between functional-specific losses.

This multi-head formulation is motivated by the following.

Data efficiency. Large datasets from inexpensive functionals may help shape shared representations, which smaller high-fidelity datasets can potentially benefit from during adaptation.

Mutual regularization. Learning multiple functionals jointly may encourage the model to capture more general structural patterns, rather than relying too heavily on dataset-specific signals.

Overall, multi-head learning provides a principled framework for cross-functional transfer, balancing shared representation learning with task-specific specialization.

4. Results

This section introduces the results of the different methods discussed earlier. The first section on delta-learning first uses linear probing to justify the design of our method. We then present the results of the delta-learning method showing its data and compute efficiency across datasets. Lastly we present our results on multi-head finetuning followed by an ablation study on delta-learning.

4.1. Delta-learning

4.1.1. Layer selection with frozen probes

We first probe frozen Orb-v3 representations across depth to select the layer used by the delta-learning head. Figure 1 shows a consistent pattern across datasets and probe families: the first message-passing layer (MP1) yields the lowest test MAE, while deeper layers perform progressively worse. This ranking is preserved for both the Ridge probe and a one-hidden-layer MLP probe, showing that the effect is not specific to a linear readout. With the MLP probe, MP1 reaches 0.0939 ± 0.0027 eV/atom on MatPES-PBE and 0.1007 ± 0.0026 eV/atom on MatPES-r2SCAN, while MP5 degrades to 0.2274 ± 0.0015 and 0.2377 ± 0.0008 eV/atom, respectively. The degradation is milder on OMat24 (Barroso-Luque et al., 2024), where MLP test MAE increases from 0.0613 ± 0.0052 at MP1 to 0.0858 ± 0.0012 eV/atom at MP5. This suggests that deeper Orb-v3 features are increasingly specialized to the pretraining regime and less reusable under cross-functional transfer. Based on this result, we use MP1 as input to the delta-learning head in the experiments below.

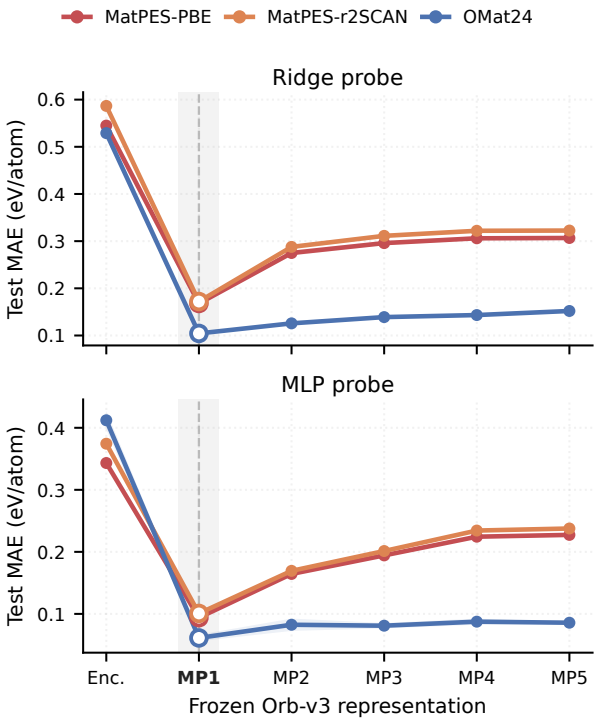

Figure 1. Layer-wise probing of frozen Orb-v3 representations. OMat24 (train) is included as a source-domain reference, while MatPES-PBE and MatPES-r2SCAN are transfer targets. Across all datasets, the first message-passing layer yields the lowest MAE for both Ridge and MLP probes, and performance deteriorates at deeper layers. The degradation is substantially stronger on the transfer targets than on the source-domain reference.

4.1.2. Efficiency of delta-learning

We next test delta-learning data efficiency on MatPES (Kaplan et al., 2025) as well as 100k randomly sampled molecules from LeMatBulk (Siron et al., 2025), we also test compute efficiency on MatPES. The data efficiency results can be found in Figure 2 for the MatPES dataset and 4 for the LeMatBulk dataset, the compute efficiency can be found in Figure 3. We observe that under 20k samples, the delta-learning method is consistently more efficient on both PBE to PBE and PBE to r2SCAN. On LeMatBulk (Figure 4), a dataset that is easier for Orb (the zero shot prediction has a 3x smaller MAE/atom), the delta learning method keeps beating full finetuning even with 100k training samples and improves significantly on the Orb baseline with only 90 training samples.

On compute efficiency, the delta-learning method usually requires around 100x less compute; the model can be easily adapted to new functionals on a CPU in less than an hour. These results suggest a wide range of applications where delta-learning can be favored over full finetuning, extending the task on which we can adapt atomistic foundation models. However, we notice that delta-learning is unable to reach full finetuning performance when large amounts of data are available (full dataset performances are shown in Table 1).

Therefore, when the functional or chemical domain is well populated with labeled data, full-finetuning or multi-task finetuning is preferable. Note that in Table 1, the delta-learning and full fine-tuning models are learned for a single task at a time (MatPES PBE *or* MatPES r2SCAN) while the multi-head is one stand-alone model trained on both. We also tested delta-learning to predict stress; we show the results in Table 3 in Appendix A.

Table 1. Full dataset performances of the different methods on MatPES. The errors are reported in eV/atom.

Method	PBE error	r2SCAN error
Delta-learning head	0.0390	0.0685
Single-task full fine-tuning	0.0262	0.0328
Multi-head fine-tuning	0.0423	0.0447

4.2. Multi-head fine-tuning

We study the quality of the transfer from PBE to r2SCAN on the same two MatPES datasets, each with 500k datapoints. We gradually increase the amount of r2SCAN used datapoints while using the whole PBE set. The transfer results can be found in Figure 5.

Naturally, the r2SCAN MAE/atom decreases when more data is used. Interestingly, we also observe a consistent improvement in PBE performance as the amount of r2SCAN data increases. This indicates that multi-head learning

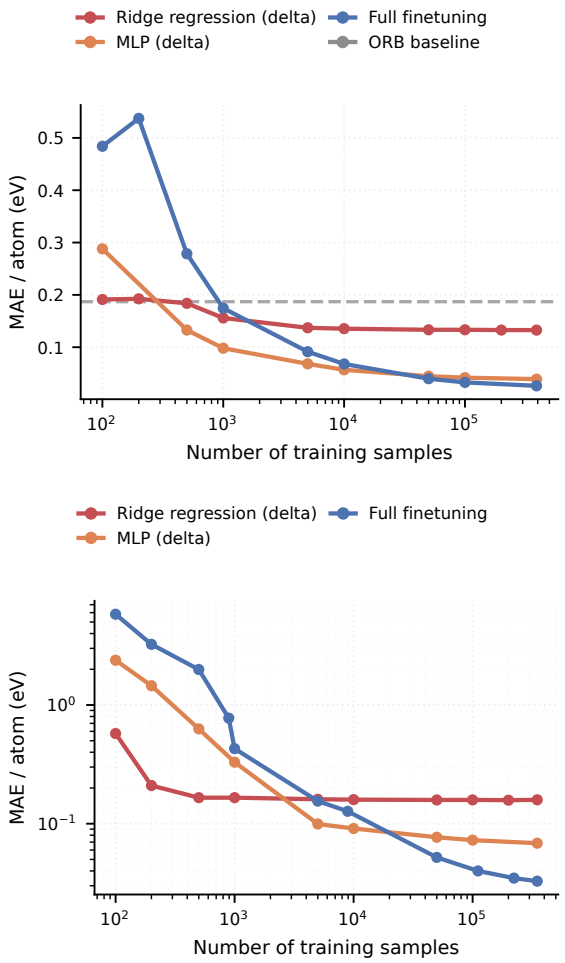

Figure 2. Top: MAE (eV/atom) as a function of training set size on MatPES PBE (data is log scale). Bottom: MAE (eV/atom) as a function of training set size on MatPES r2SCAN (log-log scale).

enables bidirectional transfer between functionals, rather than a unidirectional transfer from low- to high-precision regimes.

A possible explanation is that high-precision functionals such as r2SCAN provide cleaner and more informative supervision signals, which help refine the shared latent representation. As a result, the backbone learns features that are not only transferable to r2SCAN, but also improve generalization on PBE.

This phenomenon highlights the role of multi-task learning as a form of implicit regularization: by jointly learning multiple approximations of the same underlying physical quantity, the model is encouraged to capture more robust and physically meaningful patterns.

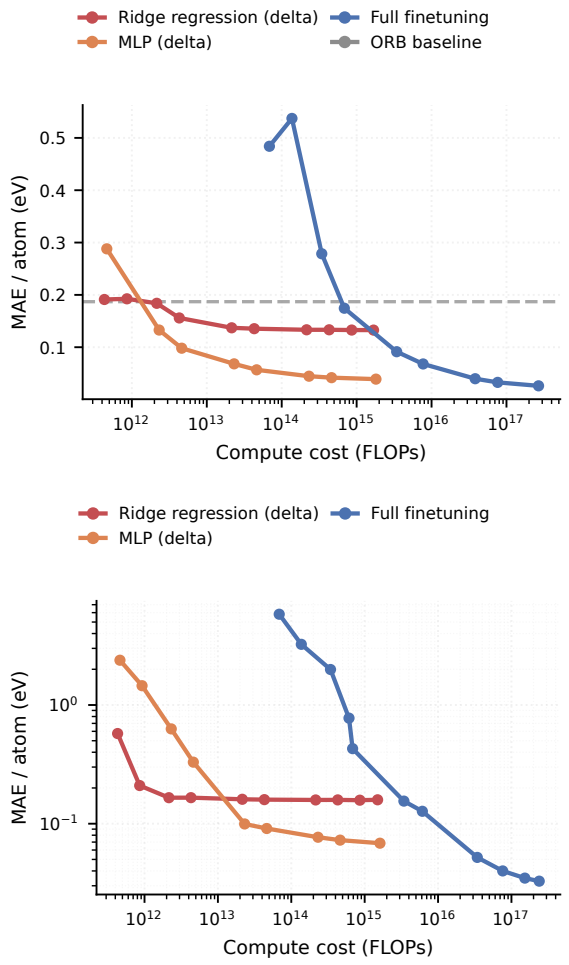

Figure 3. Top: MAE (eV/atom) as a function of compute on MatPES PBE (the flops axis is log scale). Bottom: MAE (eV/atom) as a function of compute on MatPES r2SCAN.(log-log scale).

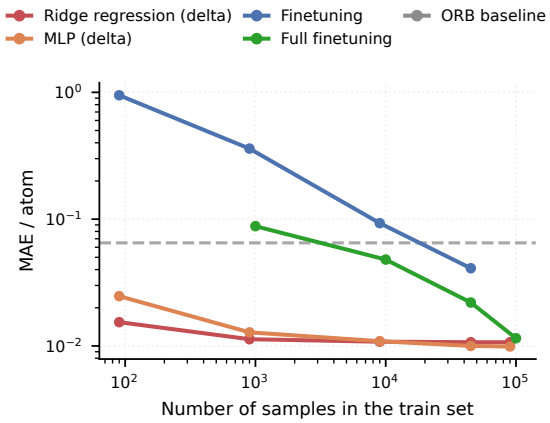

Figure 4. MAE (eV/atom) as a function of training set size on LeMatBulk PBE (log-log scale).”Finetuning performances” refers to training only the prediction head and keep the backbone frozen.

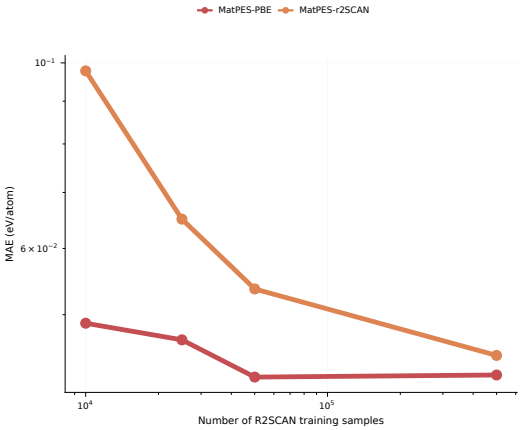

Figure 5. MAE (eV/atom) as a function of the r2SCAN training set size. With a 500k MatPES base

4.3. Ablation study

In order to validate our delta learning approach, we tested different approaches to delta learning. In Table 2, we tested different features for delta-learning : Orb baseline is the Orb model zero shot performance, ”emb” corresponds to the molecule embedding, ”pred” corresponds to the Orb energy or stress prediction and ”natoms” corresponds to the number of atoms, which we added as a feature to see if it improved performances. These results show that both the embedding and Orb’s prediction are important to improve on the model’s zero shot performances. Surprisingly, the information of Orb’s prediction on stress is enough to help the model perform similarly than with the energy prediction, which suggest that similar information are present in those two different prediction distributions. Additional results on stress to stress transfer are available in Appendix A.

Table 2. Comparison of ridge regression delta learning performances with different features on the MatPES dataset.

Method	PBE error	r2SCAN error
Orb zero-shot baseline	0.187	14.2
Embedding + Orb energy prediction	0.133	0.159
Embedding + Orb stress prediction	0.134	0.161
Embedding only	0.540	0.580
Orb energy prediction only	0.233	9.21
Embedding + Orb energy prediction + atom count	0.133	0.159

5. Conclusion

This paper introduces two new methods for adapting an atomistic foundation model to a new functional or chemical domain. Through layer-wise probing, we show that early layers of Orb-v3 are much more transferable across DFT functionals than deeper layers. This observation motivates our delta-learning method which uses frozen embeddings along with the foundation model’s original prediction. Across low-data regimes, this method outperforms

full fine-tuning with fewer than 20k diverse target samples, improves over the baseline with only 90 LeMat-Bulk samples, and typically reduces training cost by around two orders of magnitude. Complementarily, multi-head fine-tuning quantifies how supervision from one functional can improve another through a shared representation: adding r2SCAN supervision improves PBE predictions as well as r2SCAN predictions, showing that cross-functional transfer can be bidirectional. Together, these results provide a practical recipe for choosing between lightweight adaptation and joint multi-functional training under realistic data and compute constraints.

References

Barroso-Luque, L., Shuaibi, M., Fu, X., Wood, B. M., Dzamba, M., Gao, M., Rizvi, A., Zitnick, C. L., and Ulissi, Z. W. Open materials 2024 (omat24) inorganic materials dataset and models, 2024. URL https://arxiv.org/abs/2410.12771.

Batatia, I., Lin, C., Hart, J., Kasoar, E., Elena, A. M., Norwood, S. W., Wolf, T., and Csányi, G. Cross learning between electronic structure theories for unifying molecular, surface, and inorganic crystal foundation force fields. *arXiv preprint arXiv: 2510.25380*, 2025.

Deng, B., Zhong, P., Jun, K., Riebesell, J., Han, K., Barbel, C. J., and Ceder, G. CHGNet: Pretrained universal neural network potential for charge-informed atomistic modeling. *arXiv preprint arXiv:2302.14231*, 2023.

Duval, A., Mathis, S., Joshi, C. K., Schmidt, V., Miret, S., Malliaros, F. D., Cohen, T., Liò, P., Bengio, Y., and Bronstein, M. A hitchhiker's guide to geometric GNNs for 3D atomic systems. *arXiv preprint arXiv:2312.07511*, 2023.

Furness, J. W., Kaplan, A. D., Ning, J., Perdew, J. P., and Sun, J. Accurate and numerically efficient r2scan meta-generalized gradient approximation. *The Journal of Physical Chemistry Letters*, 11(19):8208–8215, 2020. doi: 10.1021/acs.jpclett.0c02405. URL https://doi.org/10.1021/acs.jpclett.0c02405. PMID: 32876454.

Hoffmann, N., Schmidt, J., Botti, S., and Marques, M. A. Transfer learning on large datasets for the accurate prediction of material properties. *Digital Discovery*, 2(5): 1368–1379, 2023.

Huang, X., Deng, B., Zhong, P., Kaplan, A. D., Persson, K. A., and Ceder, G. Cross-functional transferability in foundation machine learning interatomic potentials. *npj Computational Materials*, 11(1):313, 2025.

Hutchinson, M. L., Antono, E., Gibbons, B. M., Paradiso, S., Ling, J., and Meredig, B. Overcoming data scarcity with transfer learning. *arXiv preprint arXiv:1711.05099*, 2017.

Kaplan, A. D., Liu, R., Qi, J., Ko, T. W., Deng, B., Riebesell, J., Ceder, G., Persson, K. A., and Ong, S. P. A foundational potential energy surface dataset for materials. *arXiv preprint arXiv:2503.04070*, 2025.

Kolluru, A., Shoghi, N., Shuaibi, M., Goyal, S., Das, A., Zitnick, C. L., and Ulissi, Z. Transfer learning using attentions across atomic systems with graph neural networks (TAAG). *The Journal of Chemical Physics*, 156 (18), 2022.

Liu, S., Zou, D., Shoghi, N., Fung, V., Liu, K., and Li, P. Roft-mol: Benchmarking robust fine-tuning with molecular graph foundation models. *arXiv preprint arXiv:2509.00614*, 2025.

Perdew, J. P., Burke, K., and Ernzerhof, M. Generalized gradient approximation made simple. *Physical review letters*, 77(18):3865, 1996.

Pinto, L. Superior molecular representations from intermediate encoder layers. *arXiv preprint arXiv:2506.06443*, 2025.

Pitfield, J., Brix, F., Tang, Z., Slavensky, A. M., Rønne, N., Christiansen, M. P. V., and Hammer, B. Augmentation of universal potentials for broad applications. *Physical Review Letters*, 134(5):056201, 2025.

Radova, M., Stark, W. G., Allen, C. S., Maurer, R. J., and Bartók, A. P. Fine-tuning foundation models of materials interatomic potentials with frozen transfer learning. *npj Computational Materials*, 11(1):237, 2025.

Ramakrishnan, R., Dral, P. O., Rupp, M., and Lilienfeld, O. A. V. Big data meets quantum chemistry approximations: the δ-machine learning approach. *Journal of Chemical Theory and Computation*, 11(5):2087–2096, 2015.

Rhodes, B., Vandenhaute, S., Šimkus, V., Gin, J., Godwin, J., Duignan, T., and Neumann, M. Orb-v3: atomistic simulation at scale. *arXiv preprint arXiv:2504.06231*, 2025.

Shimakawa, H., Kumada, A., and Sato, M. Extrapolative prediction of small-data molecular property using quantum mechanics-assisted machine learning. *npj Computational Materials*, 10(1):11, 2024.

Shoghi, N., Kolluru, A., Kitchin, J. R., Ulissi, Z. W., Zitnick, C. L., and Wood, B. M. From molecules to materials: Pre-training large generalizable models for atomic property prediction. *arXiv preprint arXiv:2310.16802*, 2023.

Siron, M., Djafar, I., Ramlaoui, A., du Fayette, E., Rossello, A., Fako, E., McDermott, M., Therrien, F., Barroso-Luque, L., Cipcigan, F., Schwaller, P., Wolf, T., and Duval, A. LeMat-Bulk: aggregating, and de-duplicating quantum chemistry materials databases. *arXiv preprint arXiv:2511.05178*, 2025.

Sun, J., Ruzsinszky, A., and Perdew, J. P. Strongly constrained and appropriately normed semilocal density functional. *Physical review letters*, 115(3):036402, 2015.

Wood, B. M., Dzamba, M., Fu, X., Gao, M., Shuaibi, M., Barroso-Luque, L., Abdelmaqsoud, K., Gharakhanyan, V., Kitchin, J. R., Levine, D. S., Michel, K., Sriram, A., Cohen, T., Das, A., Sahoo, S. J., Rizvi, A., Ulissi, Z. W., and Zitnick, C. L. UMA: A family of universal models for atoms. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=SvopaNxYWt.

Zhang, C., Zhang, D., Peng, A., Guo, M., Zhang, Y., Wang, L., Ke, G., Zhang, L., Li, T., and Wang, H. Multi-task fine-tuning enables robust out-of-distribution generalization in atomistic models. *arXiv preprint arXiv:2601.08486*, 2026.

A. Results of Delta-learning on Stress

Table 3 shows the result of ridge regression delta learning on predicting stress given Orbv3's stress predictions. We report unaltered MAE on MatPES PBE and r2SCAN. The name of the features is analogous to Table 2. These results show that the delta learning method also works when trying to predict other material properties.

Table 3. Comparison of ridge regression delta learning performances with different features on the MatPES dataset.

Method	**PBE error**	**r2SCAN error**
Orb zero-shot baseline	0.0415	0.0445
Embedding + Orb stress prediction	0.0067	0.0081

B. Experimental and Training Details

B.1. Experimental Setup

All experiments use Orb-v3 (Rhodes et al., 2025) as the pretrained atomistic backbone. Energy prediction experiments are evaluated using mean absolute error (MAE). Unless stated, energy errors are reported per atom unless reported explicitly different. For probing experiments, the target is the formation energy per atom and all reported values are in eV/atom.

We evaluate transfer on MatPES-PBE and MatPES-r2SCAN (Kaplan et al., 2025). For the data-scaling experiments, we additionally evaluate on a randomly sampled subset of 100k structures from LeMatBulk (Siron et al., 2025). OMat24 (Barroso-Luque et al., 2024) is used only as a source-domain reference in the layer-wise probing analysis.

For layer-wise probing, all probe families and all probed layers are evaluated under matched formula-based train/validation/test splits. Within each split, structures with the same chemical formula are assigned to the same subset,limiting composition-level leakage between training, validation, and test evaluation. The same partition is reused across layers and probe families; for paired MatPES-PBE and MatPES-r2SCAN, it is also reused across functionals. We repeat the protocol over multiple split seeds and report mean and standard deviation across splits.

For methods that use normalization, statistics are computed on the training set only using a least squares fitting and then applied to the validation and test sets.

B.2. Delta-learning details

Delta-learning is used as a lightweight correction on top of a frozen Orb-v3 model. The backbone is first used to precompute both its scalar prediction and its latent representation for each structure. The adaptation model is then trained only on these fixed features.

For a structure $\mathcal{G}$ with n atoms, let $\mathbf{h}_i^{(\ell)}(\mathcal{G})$ denote the atom-level representation of atom i at layer ℓ of Orb-v3. We use the mean-pooled graph representation

$$\bar{\mathbf{h}}^{(\ell)}(\mathcal{G}) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{h}_i^{(\ell)}(\mathcal{G}).$$

The delta-learning model receives the concatenated feature vector

$$\mathbf{z}(\mathcal{G}) = \left[\bar{\mathbf{h}}^{(\ell^\star)}(\mathcal{G}), E_{\mathrm{Orb}}(\mathcal{G})\right],$$

where $E_{\mathrm{Orb}}(X)$ is the Orb-v3 energy prediction and $\ell^\star$ is the layer selected by the probing analysis. Based on the results in Section 4.1 and Appendix C.1, we use the first message-passing layer, MP1 for the main delta-learning experiments.

The model is trained to predict the residual between the target DFT value and the Orb-v3 prediction:

$$\Delta E(\mathcal{G}) = E_{\mathrm{DFT}}(\mathcal{G}) - E_{\mathrm{Orb}}(\mathcal{G}).$$

The corrected prediction is therefore

$$E_{\mathrm{pred}}(X) = E_{\mathrm{Orb}}(X) + E_{\Delta}(z(X)).$$

We evaluate two delta-learning models. The first is Ridge regression, where the solution is computed analytically and the regularization coefficient is selected on the validation set. The second is a multilayer perceptron (MLP) trained with Adam using a learning rate of 3×10^{-4}. The MLP is trained for 500 epochs with a batch size of 32.

The main compute advantage of this approach comes from freezing Orb-v3 and precomputing its predictions and embeddings once for each dataset. After this preprocessing step, training the delta-learning model requires updating only a small regression model rather than the full atomistic backbone.

B.3. Layer-wise probing details

Layer-wise probing is used to identify which frozen Orb-v3 representations are most reusable under cross-functional transfer. All Orb-v3 parameters are kept fixed; only the probe parameters are trained. We evaluate representations from the encoder output and from the five successive message-passing blocks. We denote the encoder output by layer 0 and the message-passing blocks by MP1–MP5.

For a structure $\mathcal{G}$ with n atoms, let $\mathbf{h}_i^{(\ell)}(\mathcal{G})$ be the atom-level representation of atom i at layer ℓ. We obtain a graph-level representation by mean pooling,

$$\mathbf{z}_\ell(\mathcal{G}) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{h}_i^{(\ell)}(\mathcal{G}).$$

In addition to Orb-v3 representations, we evaluate a composition-only baseline built from normalized elemental fractions. This baseline contains stoichiometric information but no learned structural representation.

For each layer, we train a supervised probe

$$f_\ell : \mathbf{z}_\ell(\mathcal{G}) \mapsto \hat{y},$$

where y is the formation energy per atom.

For MatPES, the PBE and r2SCAN subsets are exactly paired structure by structure, with matched identifiers and formulas. For each split seed $s \in \{11, 22, 33\}$, we construct formula-based train/validation/test partitions,

$$\mathcal{D} = \mathcal{D}_{\text{train}}^{(s)} \cup \mathcal{D}_{\text{val}}^{(s)} \cup \mathcal{D}_{\text{test}}^{(s)},$$

using 70%, 15%, and 15% of the data, respectively. All structures with the same chemical formula are assigned to the same subset. Within each split seed, the same partition is used for all layers and probe families. For paired MatPES-PBE and MatPES-r2SCAN, the same partition is also shared across both functionals. This avoids composition-level leakage and ensures that differences across layers are not driven by different train/test compositions.

Inputs and targets are standardized using statistics computed on $\mathcal{D}_{\text{train}}^{(s)}$ only. The same transformations are then applied to validation and test data. All reported errors are in eV/atom.

We evaluate two probe families. The first is a Ridge probe,

$$f_\ell^{\text{ridge}}(\mathbf{z}) = \mathbf{w}_\ell^\top \mathbf{z} + b_\ell,$$

with the regularization coefficient selected on the validation split from

$$\alpha \in \{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 100, 1000\}.$$

The second is a one-hidden-layer MLP,

$$f_\ell^{\text{mlp}}(\mathbf{z}) = \mathbf{W}_{2,\ell}\, \sigma(\mathbf{W}_{1,\ell}\mathbf{z} + \mathbf{b}_{1,\ell}) + b_{2,\ell},$$

where σ is the SiLU activation and the hidden dimension is 256. The MLP is trained with Adam using mini-batches of size 1024 for at most 750 epochs. The learning rate and weight decay are selected on the validation split from

$$\eta \in \{10^{-4}, 3 \times 10^{-4}, 10^{-3}, 3 \times 10^{-3}\}, \qquad \lambda \in \{10^{-4}, 10^{-3}, 10^{-4}\}.$$

Validation MAE is evaluated every 10 epochs, early stopping uses patience 20, and each MLP configuration is run with 3 random restarts. The restart with the lowest validation MAE is retained.

For each layer, split seed, and probe family, hyperparameters are selected as the configuration with the lowest validation MAE. After hyperparameter selection, the probe is refit on $\mathcal{D}^{(s)}_{\text{train}} \cup \mathcal{D}^{(s)}_{\text{val}}$ using the selected hyperparameters and evaluated once on the held-out test set:

$$\text{MAE}^{(s)}_{\ell} = \frac{1}{|\mathcal{D}^{(s)}_{\text{test}}|} \sum_{(\mathcal{G},y)\in\mathcal{D}^{(s)}_{\text{test}}} |f_{\ell}(\mathbf{z}_{\ell}(\mathcal{G})) - y| \,.$$

Final results are reported as the mean and standard deviation of $\text{MAE}^{(s)}_{\ell}$ across the three split seeds.

B.4. Multi-head fine-tuning details

Multi-head fine-tuning uses a single Orb-v3 backbone shared across functionals and a separate prediction head for each functional. In the experiments in this paper, the task set is

$$\mathcal{K} = \{\text{PBE}, \text{r2SCAN}\}.$$

For each structure $\mathcal{G}$, the shared backbone computes a graph representation $\mathbf{h}_{\mathcal{G}}$ and each functional k has its own prediction head:

$$E^{(k)}_{\text{pred}}(\mathcal{G}) = \text{Head}_k(\mathbf{h}_{\mathcal{G}}).$$

The multi-head model is trained with a weighted sum of per-functional losses:

$$\mathcal{L} = \sum_{k\in\mathcal{K}} \lambda_k \mathcal{L}_k.$$

Each loss $\mathcal{L}_k$ is computed only on samples with labels for functional k. This allows the model to use datasets with different numbers of labels per functional.

Because different functionals can have different energy offsets and variances, we normalize targets separately for each functional by fitting the reference energies on each task.

This setup separates shared representation learning from functional-specific calibration. The shared backbone can exploit structural information common across functionals, while the task-specific heads can learn functional-dependent shifts in scale, offset, and target geometry.

C. Additional Results for Layer-wise Probing

This appendix provides the detailed results underlying the layer-wise probing analysis presented in Section 4.1. We report the full per-layer results for both linear (Ridge) and nonlinear (MLP) probes, and provide an additional linearity-gap analysis.

C.1. Detailed per-layer probe performance

Tables 4 and 5 report the full probe results for all evaluated representations. In addition to the Orb-v3 layers, we include a composition-only baseline and the encoder output. The composition-only baseline represents each structure by its normalized elemental fractions and therefore contains stoichiometric information only, without geometric or learned structural features. Three conclusions are consistent across probe families. First, the first message-passing layer (MP1) has the lowest test MAE on all three datasets. Second, both the composition-only baseline and the encoder output perform worse, showing that the gain is not explained by stoichiometry alone or by a purely pre-message-passing representation. Third, performance degrades at deeper layers, with a stronger degradation on the transfer targets than on the source-domain reference, OMat24 (train).

The magnitude of the depth-dependent degradation differs across domains. Under the MLP probe, the error increases from 0.0939 ± 0.0027 eV/atom at MP1 to 0.2274 ± 0.0015 eV/atom at MP5 on MatPES-PBE, and from 0.1007 ± 0.0026 to 0.2377 ± 0.0008 eV/atom on MatPES-r2SCAN. On OMat24, the increase is milder, from 0.0613 ± 0.0052 to 0.0858 ± 0.0012 eV/atom. The same ordering is observed with the Ridge probe. This supports the conclusion that MP1 is the most reusable frozen representation, while deeper representations degrade more strongly under cross-functional transfer.

Table 4. Detailed Ridge-probe test MAE (eV/atom) across layers and datasets. Values are mean ± standard deviation across three formula-based split seeds.

Layer	MatPES (PBE)	MatPES (r2SCAN)	OMat24 (train)
Composition	0.5450 ± 0.0034	0.5867 ± 0.0050	0.5290 ± 0.0009
Encoder	0.5450 ± 0.0031	0.5867 ± 0.0046	0.5290 ± 0.0009
MP1	0.1679 ± 0.0020	0.1720 ± 0.0026	0.1044 ± 0.0011
MP2	0.2749 ± 0.0058	0.2878 ± 0.0049	0.1257 ± 0.0006
MP3	0.2959 ± 0.0042	0.3112 ± 0.0039	0.1390 ± 0.0005
MP4	0.3061 ± 0.0044	0.3220 ± 0.0045	0.1433 ± 0.0010
MP5	0.3067 ± 0.0041	0.3224 ± 0.0040	0.1520 ± 0.0013

Table 5. Detailed MLP-probe test MAE (eV/atom) across layers and datasets..

Layer	MatPES (PBE)	MatPES (r2SCAN)	OMat24 (train)
Composition	0.3682 ± 0.0015	0.3855 ± 0.0036	0.4078 ± 0.0032
Encoder	0.3432 ± 0.0081	0.3744 ± 0.0033	0.4121 ± 0.0150
MP1	0.0939 ± 0.0027	0.1007 ± 0.0026	0.0613 ± 0.0052
MP2	0.1644 ± 0.0025	0.1694 ± 0.0026	0.0825 ± 0.0100
MP3	0.1943 ± 0.0023	0.2013 ± 0.0024	0.0810 ± 0.0044
MP4	0.2244 ± 0.0015	0.2343 ± 0.0005	0.0875 ± 0.0050
MP5	0.2274 ± 0.0015	0.2377 ± 0.0008	0.0858 ± 0.0012

C.2. Functional gap in paired MatPES probing

Because MatPES-PBE and MatPES-r2SCAN are paired structure by structure, we can compare their probing errors at fixed representation depth. We define the functional probing gap as

$$\Delta^{(\ell)}_{\mathrm{r2SCAN-PBE}} = \mathrm{MAE}^{(\ell)}_{\mathrm{r2SCAN}} - \mathrm{MAE}^{(\ell)}_{\mathrm{PBE}}.$$

Positive values indicate that r2SCAN is harder to predict than PBE from the same frozen representation.

Table 6. Functional probing gap on paired MatPES structures, reported as $\mathrm{MAE}_{\mathrm{r2SCAN}} - \mathrm{MAE}_{\mathrm{PBE}}$ in eV/atom. Values are means across three formula-based split seeds.

Layer	Ridge gap	MLP gap
Composition	0.0417	0.0173
Encoder	0.0417	0.0312
MP1	0.0041	0.0069
MP2	0.0129	0.0050
MP3	0.0153	0.0070
MP4	0.0159	0.0099
MP5	0.0157	0.0103

The gap is positive for all layers and both probe families, indicating that r2SCAN is consistently slightly harder to predict than PBE from the same frozen Orb-v3 representation. This is expected because Orb-v3 is pretrained on the OMat24 source domain, which is closer to PBE-level energetics than to r2SCAN. However, the magnitude of this functional gap remains small compared with the depth-dependent degradation. With the MLP probe, the r2SCAN–PBE gap is 0.0069 eV/atom at MP1, whereas the MP1-to-MP5 degradation is 0.1335 eV/atom on PBE and 0.1370 eV/atom on r2SCAN.

This suggests that frozen Orb-v3 representations retain broadly reusable structural and compositional information for both MatPES functionals, despite the pretraining bias toward the OMat24 source domain. The main limitation revealed by probing is therefore not a large loss of information specific to r2SCAN, but the progressive specialization of deeper Orb-v3 layers to the pretraining regime. Consequently, representation depth is the dominant factor in the probing analysis, with MP1 providing the best trade-off between learned chemical structure and transferability.