# MoMa: A Modular Deep Learning Framework for Material Property Prediction

**Botian Wang**[1,2*] **Yawen Ouyang**[1*] **Yaohui Li**[3*] **Yiqun Wang**[1] **Haorui Cui**[2]
**Jianbing Zhang**[3] **Xiaonan Wang**[4] **Wei-Ying Ma**[1] **Hao Zhou**[1]
[1] Institute for AI Industry Research (AIR), Tsinghua University
[2] Department of Computer Science and Technology, Tsinghua University
[3] School of Artificial Intelligence, Nanjing University &
National Key Laboratory for Novel Software Technology, Nanjing University
[4] Department of Chemical Engineering, Tsinghua University
wbt23@mails.tsinghua.edu.cn {maweiying,zhouhao}@air.tsinghua.edu

## Abstract

Deep learning methods for material property prediction have been widely explored to advance materials discovery. However, the prevailing pre-train then fine-tune paradigm often fails to address the inherent diversity and disparity of material tasks. To overcome these challenges, we introduce MoMa, a **Mo**dular framework for **Ma**terials that first trains specialized modules across a wide range of tasks and then adaptively composes synergistic modules tailored to each downstream scenario. Evaluation across 17 datasets demonstrates the superiority of MoMa, with a substantial 14% average improvement over the strongest baseline. Few-shot and continual learning experiments further highlight MoMa's potential for real-world applications. Pioneering a new paradigm of modular material learning, MoMa will be open-sourced to foster broader community collaboration.

## 1 Introduction

Accurate and efficient material property prediction is critical for accelerating materials discovery. Key properties such as formation energy and band gap play fundamental roles in identifying stable materials and functional semiconductors (Riebesell et al., 2023; Masood et al., 2023). While traditional approaches such as density functional theory (DFT) offer high precision, their prohibitive computational cost limits their practicality for large-scale screening (Fiedler et al., 2022).

Recently, deep learning methods have been developed to expedite traditional approaches (Xie & Grossman, 2018; Griesemer et al., 2023). Pre-trained force field models, in particular, have shown remarkable success in generalizing to a wide spectrum of material property prediction tasks (Yang et al., 2024b; Barroso-Luque et al., 2024; Shoghi et al., 2023), outperforming specialized models trained from scratch. These models are typically pre-trained on the potential energy surface (PES) data of materials and then fine-tuned for the target downstream task.

Despite these advances, we identify two key challenges that undermine the effectiveness of current pre-training strategies for material property prediction: **diversity** and **disparity**.

First, material tasks exhibit significant diversity (Fig. 1), which current pre-trained models fail to adequately cover. Existing models trained on PES-derived properties (e.g., force, energy and stress) mostly focus on crystalline materials (Yang et al., 2024b; Barroso-Luque et al., 2024). However, material tasks span wide variety of systems (e.g., crystals, organic molecules) and properties (e.g., thermal stability, electronic behavior, mechanical strength), making it difficult for methods trained on a limited set of data to generalize across the full spectrum of tasks.

Second, the disparate nature of material tasks presents huge obstacles for jointly pre-training a broad span of tasks. Material systems vary significantly in terms of bonding, atomic composition, and structural periodicity, while their properties are governed by distinct physical laws. For example,

*Equal Contribution. Correspondence to Hao Zhou (zhouhao@air.tsinghua.edu).

mechanical strength in metals is primarily influenced by atomic bonding and crystal structure, whereas electronic properties like conductivity are determined by the material's electronic structure and quantum mechanics. Consequently, training a single model across a wide range of tasks (Shoghi et al., 2023) may lead to knowledge conflicts, hindering the model's ability to effectively adapt to downstream scenarios.

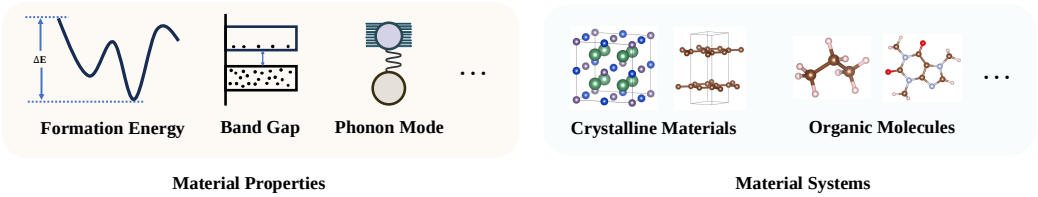


Figure 1: Illustration of the diversity of material properties (left) and systems (right). Note that material tasks are also disparate, with different laws governing the diverse properties and systems. These characteristics pose challenges for pre-training material property prediction models.

In this paper, we propose MoMa, a **Mo**dular deep learning framework for **Ma**terial property prediction, to address the diversity and disparity challenge. To accommodate the **diversity** of material tasks, MoMa first trains on a multitude of high-resource property prediction datasets, centralizing them into transferrable modules. Furthermore, MoMa incorporates an adaptive composition algorithm that customizes support for diverse downstream scenarios. Recognizing the **disparity** among material tasks, MoMa encapsulates each task within a specialized module, eliminating task interference of joint training. In adapting MoMa to specific downstream tasks, its composition strategy adaptively integrates only the most synergistic modules, mitigating knowledge conflicts and promoting positive transfer.

Specifically, MoMa comprises two major stages: (1) *Module Training & Centralization*. Drawing inspiration from modular deep learning (Pfeiffer et al., 2023), MoMa trains dedicated modules for a broad range of material tasks, offering two versions: a full module for superior performance and a memory-efficient adapter module. These trained modules are centralized in MoMa Hub, a repository designed to facilitate knowledge reuse while preserving proprietary data for privacy-aware material learning. (2) *Adaptive Module Composition* (AMC). MoMa introduces the data-driven AMC algorithm that composes synergetic modules from MoMa Hub. AMC first estimates the performance of each module on the target task in a training-free manner, then heuristically optimizes their weighted combination. The resulting composed module is then fine-tuned for improved adaptation to the downstream task. Together, the two stages deliver a modular solution that enables MoMa to account for the diversity and disparity of material knowledge.

Empirical results across 17 downstream tasks showcase the superiority of MoMa, outperforming all baselines in **16/17** tasks, with an average improvement of **14%** compared to the second-best baseline. In **few-shot** settings, which are common in materials science, MoMa achieves even larger performance gains to the conventional pre-train then fine-tune paradigm. Additionally, we show that MoMa can expand its capability in **continual learning** settings by incorporating molecular tasks into MoMa Hub. The trained modules in MoMa Hub will be open-sourced, and we envision MoMa becoming a pivotal platform for the modularization and distribution of materials knowledge, fostering deeper community engagement to accelerate materials discovery.

## 2 METHOD

MoMa is a simple modular framework targeting the diversity and disparity of material tasks. The predominant pre-train then fine-tune strategy can only leverage a limited range of interrelated source tasks or indiscriminately consolidating conflicting knowledge into one model, resulting in suboptimal downstream performance. In contrast, the modular design of MoMa allows for the flexible and scalable integration of diverse material knowledge modules, and the effective and tailored adaptation to material property prediction tasks. Fig. 2 illustrates this comparison.

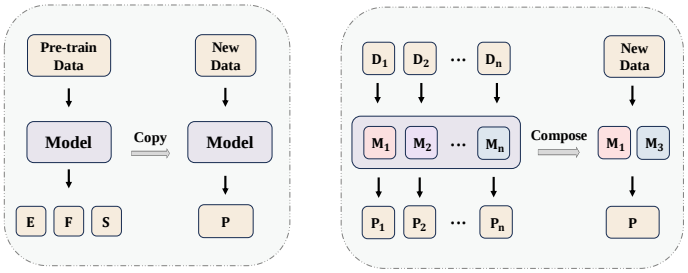


Figure 2: A comparison between the pre-train fine-tune paradigm and MoMa's modular framework. (left): The prevailing scheme involves pre-training on force field data (with supervised prediction on energy, force and stress), and then transfer to downstream tasks. (right): The modular learning scheme in MoMa train and store a broad spectrum of material tasks as modules, and adaptively compose them given a new material property prediction task.

### 2.1 Overview

MoMa involves two major stages: (1) training and centralizing modules into MoMa Hub; (2) adaptively composing these modules to support downstream material tasks.

In the first stage (Sec. 2.2), we encompass a wide range of material properties and systems into MoMa Hub. This accommodates the diversity of material tasks and addresses the task disparity by training specialized module for each.

In the second stage (Sec. 2.3), we devise the Adaptive Module Composition algorithm. Given the downstream material task, the algorithm heuristically optimizes the optimal combination of module weights for MoMa Hub, and composes a customized module based on the weights, which is subsequently fine-tuned on the task for better adaptation. Respecting the diverse and disparate nature of material tasks, our adaptive approach automatically discovers synergistic modules and excludes conflicting combinations by the data-driven assignment of module weights.

A visual overview of MoMa is provided in Figure 3.

### 2.2 Module Training & Centralization

To better exploit the transferrable knowledge of open-source material property prediction datasets, we first train distinctive modules for each high-resource material task, and subsequently centralize these modules to constitute MoMa Hub.

**Module Training** Leveraging the power of state-of-the-art material property prediction models, we choose to employ a pre-trained backbone encoder $f$ as the initialization for training each MoMa module. Note that MoMa is independent of the backbone model choice, which enables smooth integration with other pre-trained backbones.

We provide two parametrizations for the MoMa modules: the full module and the adapter module. For the full module, we directly treat each fully fine-tuned backbone as a module. The adapter module serves as a parameter-efficient variant where adapter layers (Houlsby et al., 2019) are inserted between each layer of the pre-trained backbone. The adapters are updated and the rest of the backbone is frozen. All of the adapters for each task are treated as one module. This implementation trade-offs the downstream performance for a significantly lower GPU memory cost during training, which shall be favorable when the computational resource is limited. When the training converges, we store the module parameters into a centralized repository $\mathcal{H}$ termed MoMa Hub, formally:

$$\mathcal{H} = \{g_1, g_2, \ldots, g_N\}, \quad g_i = \begin{cases} \theta_f^i & \text{(full module)} \\ \Delta_f^i & \text{(adapter module)} \end{cases}$$

where $\theta_f^i$ and $\Delta_f^i$ denote the full and adapter module parameters related to the $i^{\text{th}}$ task and encoder $f$.

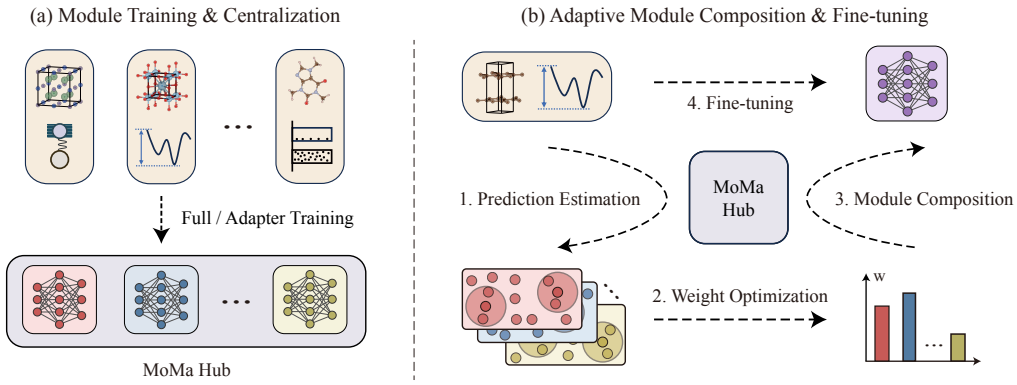


Figure 3: The MoMa framework. (a) During the Module Training & Centralization stage (Sec. 2.2), MoMa trains full and adapter modules for a wide spectrum of material tasks, constituting the MoMa Hub; (b) The Adaptive Module Composition (AMC) & Fine-tuning stage (Sec. 2.3) leverages the modules in MoMa Hub to compose a tailored module for each downstream task. The AMC algorithm comprises three steps: 1. module prediction estimation (with $k$NN); 2. module weight optimization; 3. module composition. The composed module is further fine-tuned on the task for better adaptation.

**Module Centralization** To support a wide array of downstream tasks, it is important for MoMa Hub to include modules trained on diverse material systems and properties. Currently, MoMa Hub encompasses 18 material property prediction tasks selected from the Matminer datasets (Ward et al., 2018) with over 10000 data points. These tasks span across a large range of material properties, including thermal properties (e.g. formation energy), electronic properties (e.g. band gap), mechanical properties (e.g. shear modulus) etc. For more details, please refer to Appendix B.1. To showcase the effect of scaling data diversity, we present the continual learning results in Sec. 3.5 after further incorporating molecular property prediction tasks into MoMa Hub. Note that MoMa is designed to be task-agnostic and may readily support a larger spectrum of tasks in the future.

An important benefit of the modular design of MoMa Hub is that it preserves proprietary data, which is prevalent in the field of materials, enabling privacy-aware contribution of new modules. Therefore, MoMa could serve as an open platform for the modularization of materials knowledge, which also facilitates downstream adaptation through a novel composition mechanism, as discussed in the following section.

### 2.3 Adaptive Module Composition & Fine-tuning

Given a labeled material property prediction dataset $\mathcal{D}$ with $m$ instances: $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$, the second stage of MoMa customizes a task-specific model using the modules in MoMa Hub.

To achieve this, we devise the Adaptive Module Composition (AMC) algorithm. We highlight its key desiderata:

- **Selective:** Material tasks are inherently disparate. Hence only the most relevant modules shall be selected to avoid the negative interference of materials knowledge and encourage positive transfer to downstream tasks.
- **Data-driven:** As the diversity of tasks in MoMa Hub expands, it is impossible to solely rely on human expertise for module selection. Data-driven approach is required to mine the implicit relationships between the MoMa Hub modules and downstream tasks.
- **Efficient:** Enumerating all combinations of modules is impractical. Efficient algorithms shall be developed to return the optimal module composition using a reasonable amount of computational resource.

To meet these requirements, AMC is designed as a fast heuristic algorithm that first estimates the prediction of each module on the downstream task, then optimizes the module weights, and finally

composes the selected modules to form the task-specific module. We now elaborate on the details of AMC, with its formal formulation in Algorithm 1.

**Module Prediction Estimation** We begin by estimating the predictive performance of each module in MoMa Hub $\mathcal{H}$ on the downstream task $\mathcal{D}$. More accurate predictions indicate stronger relevance to the task and intuitively warrant higher weights in the composition.

For each module $g_j$ in $\mathcal{H}$, we first take it to encode each input materials in the train set of task $\mathcal{D}$ into a set of representation $\mathcal{X}^j = \{\mathbf{x}_1^j, \mathbf{x}_2^j, \ldots, \mathbf{x}_m^j\}$ in which $\mathbf{x}_i^j = g_j(x_i)$. Then we obtain the estimated prediction of $g_j$ on $\mathcal{D}$ using a leave-one-out label propagation approach (Iscen et al., 2019). Specifically, we iteratively select one sample $\mathbf{x}_i^j$ from $\mathcal{X}^j$ and get the predicted label $\hat{y_i}^j$ by calculating the weighted sum of its $K$ nearest neighbors' labels within $\mathcal{X}^j$:

$$\hat{y_i}^j = \sum_{k=1}^{K} \frac{f_d(\mathbf{x}_i^j, \mathbf{x}_k^j)}{Z} y_k, \tag{1}$$

where $\mathbf{x}_k^j$ denotes the $k$-th nearest neighbors of $\mathbf{x}_i^j$. The distance function $f_d$ for calculating $k$NN is the exponential of cosine similarity between each pair of $\mathbf{x}_i^j$ and $\mathbf{x}_k^j$. $Z = \sum_{k=1}^{K} f_d(\mathbf{x}_i^j, \mathbf{x}_k^j)$ is the normalizing term.

While other predictors are viable, we choose $k$NN due to its good trade-off in efficiency and accuracy. Also its training-free nature enhances its flexibility in real-world scenarios, where the downstream data may be subject to updates.

**Module Weight Optimization** After estimating each module's prediction, we now have to select the optimal combination of modules tailored for the downstream task $\mathcal{D}$. To achieve this, the most straightforward approach is to compare the prediction error obtained after fine-tuning each combination of modules. However, this is infeasible due to the combinatorial explosion. Therefore, we reformulate the task as an optimization problem, using the prediction error before fine-tuning as a proxy metric (later referred to as *proxy error*). By optimizing the proxy error, we could obtain the optimal combination of weights.

Specifically, inspired by ensemble learning (Zhou et al., 2002; Zhou, 2016), we assign a weight $w_j$ for each module $g_j$ and calculate the output of the ensemble: $\sum_{j=1}^{NT} w_j \hat{y_i}^j$. We then estimate the proxy error on the train set of $\mathcal{D}$ for this weighted ensemble:

$$E_{\mathcal{D}} = \frac{1}{m} \sum_{i=1}^{m} (\sum_{j=1}^{N} w_j \hat{y}_i^j - y_i)^2 \tag{2}$$

To minimize the proxy error $E_{\mathcal{D}}$, we then utilize the open source cvxpy package (Diamond et al., 2014) to optimize the module weights. The objective is:

$$\underset{w_j}{\operatorname{argmin}}\ E_{\mathcal{D}}, \ \text{ s.t. } \sum_{j=1}^{N} w_j = 1,\ w_j \geq 0 \tag{3}$$

**Module Composition** After the optimization converges, we can use the learned weights to compose a single customized module for the specific task. It is intuitive to retain the knowledge for modules with high weights, as they are more relevant to the downstream task, while discarding the modules with zero weights, as they do not contribute in lowering the proxy error.

Inspired by the recent success of model merging in NLP and CV (Wortsman et al., 2022; Ilharco et al., 2022; Yu et al., 2024; Li et al., 2024; Yang et al., 2024a), we adopt a simple yet surprisingly effective method by weighted averaging the parameters of the selected modules:

$$g_{\mathcal{D}} = \sum_{j=1}^{N} w_j^* g_j, \tag{4}$$

where $w_j^*$ represents the optimized weight for the $j$-th module in Eq. (3). Here, the weights underscore the relevance of each selected module to the downstream task.

Opting for a weighted average over a simple average allows the composed module to focus on exploiting the most relevant aspects of materials knowledge, delivering empirical benefits as evidenced by our ablation study (Sec. 3.3). For the full module parametrization, all modules share the same architecture with the pre-trained backbone and have identical initializations, so it paves way for the successful composition of module knowledge (Zhou et al., 2024).

Table 1: **Main results for 17 material property prediction tasks.** The best MAE for each task is highlighted in **bold** and the second best result is underlined. Lower values indicate better performance. The results presented for each task are the average of five data splits, reported to three significant digits. For each method, the standard deviation of the test MAE across five random seeds is shown in parentheses. Additionally, the average rank and its standard deviation across the 17 datasets are provided to reflect the consistency of each method.

| Datasets | CGCNN | MoE-(18) | JMP-MT | JMP-FT | MoMa (Adapter) | MoMa (Full) |
|---|---|---|---|---|---|---|
| Experimental Band Gap (eV) | 0.471 (0.008) | 0.374 (0.008) | 0.377 (0.005) | 0.358 (0.014) | 0.359 (0.009) | **0.305** (0.006) |
| Formation Enthalpy (eV/atom) | 0.193 (0.015) | 0.0949 (0.0016) | 0.134 (0.001) | 0.168 (0.007) | 0.158 (0.009) | **0.0839** (0.0013) |
| 2D Dielectric Constant | 2.90 (0.12) | 2.29 (0.01) | 2.25 (0.06) | 2.35 (0.07) | 2.31 (0.04) | **1.89** (0.03) |
| 2D Formation Energy (eV/atom) | 0.169 (0.006) | 0.106 (0.005) | 0.140 (0.004) | 0.125 (0.006) | 0.112 (0.002) | **0.0495** (0.0015) |
| Exfoliation Energy (meV/atom) | 59.7 (1.5) | 52.5 (0.8) | 42.3 (0.5) | **35.4** (2.0) | **35.4** (0.9) | 36.3 (0.2) |
| 2D Band Gap (eV) | 0.686 (0.034) | 0.532 (0.008) | 0.546 (0.020) | 0.582 (0.018) | 0.552 (0.014) | **0.375** (0.006) |
| 3D Poly Electronic | 32.5 (1.1) | 27.7 (0.1) | 23.9 (0.2) | 23.3 (0.3) | 23.3 (0.2) | **23.0** (0.1) |
| 3D Band Gap (eV) | 0.492 (0.008) | 0.361 (0.003) | 0.423 (0.004) | 0.249 (0.001) | 0.245 (0.002) | **0.200** (0.001) |
| Refractive Index | 0.0866 (0.0014) | 0.0785 (0.0004) | 0.0636 (0.0006) | 0.0555 (0.0027) | 0.0533 (0.0023) | **0.0523** (0.0010) |
| Elastic Anisotropy | 3.65 (0.11) | 3.010 (0.03) | 2.53 (0.26) | **2.42** (0.36) | 2.57 (0.61) | 2.86 (0.28) |
| Electronic Dielectric Constant | 0.168 (0.002) | 0.157 (0.015) | 0.137 (0.002) | 0.108 (0.002) | 0.106 (0.002) | **0.0885** (0.0048) |
| Dielectric Constant | 0.258 (0.008) | 0.236 (0.002) | 0.224 (0.004) | 0.171 (0.002) | 0.168 (0.002) | **0.158** (0.002) |
| Phonons Mode Peak ($cm^{-1}$) | 0.127 (0.004) | 0.0996 (0.0083) | 0.0859 (0.0006) | 0.0596 (0.0065) | 0.0568 (0.0009) | **0.0484** (0.0026) |
| Poisson Ratio | 0.0326 (0.0001) | 0.0292 (0.0001) | 0.0297 (0.0003) | 0.0221 (0.0004) | 0.0220 (0.0003) | **0.0204** (0.0002) |
| Poly Electronic | 2.97 (0.10) | 2.61 (0.13) | 2.42 (0.03) | 2.11 (0.04) | 2.13 (0.03) | **2.09** (0.03) |
| Poly Total | 6.54 (0.24) | 5.51 (0.04) | 5.52 (0.03) | 4.89 (0.06) | 4.89 (0.04) | **4.86** (0.07) |
| Piezoelectric Modulus | 0.232 (0.004) | 0.208 (0.003) | 0.199 (0.002) | 0.174 (0.004) | **0.173** (0.003) | 0.174 (0.001) |
| **Average Rank** | 6.00 (0.00) | 4.12 (1.17) | 3.94 (0.97) | 2.88 (1.27) | 2.47 (0.94) | **1.35** (0.86) |

**Downstream Fine-tuning** To better adapt to the downstream task $\mathcal{D}$, the composed module $g_{\mathcal{D}}$ is appended with a task-specific head and then fine-tuned on $\mathcal{D}$ to convergence.

## 3 EXPERIMENTS

In this section, we conduct comprehensive experiments to showcase the empirical effectiveness of MoMa. The experimental setup is described in Sec. 3.1. The main results, presented in Sec. 3.2, show that MoMa **substantially outperforms** baseline methods. Additionally, we perform a thorough ablation study on the AMC algorithm as detailed in Sec. 3.3. In face of the data scarcity challenge common in real-world materials discovery settings, we evaluate MoMa's few-shot learning ability in Sec. 3.4, where it achieves **even larger** performance gains as compared to baselines. To further highlight the **flexibility and scalability** of MoMa, we extend MoMa Hub to include molecular datasets and present the continual learning results in Sec. 3.5. Finally, we visualize the module weights optimized by AMC in Sec. 3.6, showcasing MoMa's potential for providing **valuable insights** into material properties.

### 3.1 SETUP

We conduct experiments MoMa on 17 material property prediction tasks adhering to the benchmark settings established by Chang et al. (2022). For the backbone of MoMa, we choose to employ the JMP model (Shoghi et al., 2023). We report the mean absolute error (MAE) averaged for five random data splits as evaluation metric.

We compare the performance of MoMa with four baseline methods: CGCNN (Xie & Grossman, 2018), MoE-(18) (Chang et al., 2022), JMP-FT (Shoghi et al., 2023), and JMP-MT (Sanyal et al., 2018). CGCNN represents a classical method without pre-training. MoE-(18) trains separate CGCNN models on the upstream tasks of MoMa and ensemble them as one model in a mixture-of-experts approach for downstream fine-tuning. JMP-FT directly fine-tunes on the downstream tasks with the JMP pre-trained checkpoint. JMP-MT trains all tasks in MoMa with a multi-task pretraining scheme and then adapt to each downstream datasets with further fine-tuning.

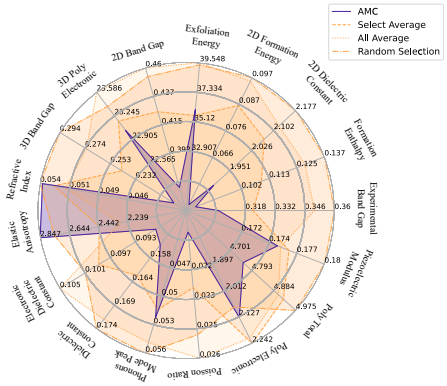


Figure 4: Ablation study of AMC. The main results using AMC (purple) are compared with the ablated variants (orange) that substitute AMC with select average, all average and random selection. The axis represents the MAE on each dataset and **smaller area is better**.

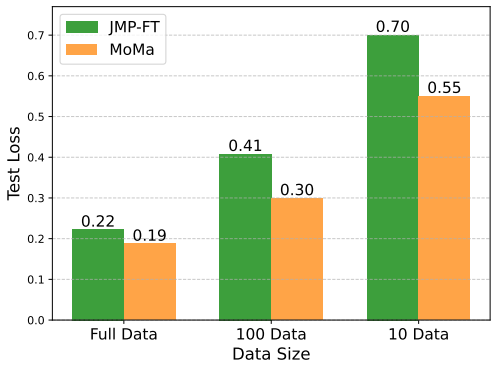


Figure 5: The average test losses of MoMa and JMP-FT across 17 downstream tasks under varying data availability settings. MoMa consistently outperforms JMP-FT in all settings. The loss reduction amplifies as the data size shrinks, showing the advantage of MoMa in few-shot settings.

More details on datasets and implementation, as well as a thorough discussion on baselines are included in Appendix B.

### 3.2 Main Results

**Performance of MoMa** As shown in Tab. 1, the MoMa (Full) achieves the best performance with the lowest average rank of 1.35 and 14/17 best results. The adapter variant of MoMa follows with an average rank of 2.47. Together, the two variants hold 16 out of 17 best results. They also exhibit the smallest rank deviations, indicating that MoMa consistently delivers reliable performance across tasks. Notably, MoMa (Full) outperforms JMP-FT in 14 tasks, with an impressive average improvement of 14.0%, demonstrating the effectiveness of MoMa Hub modules in fostering material property prediction tasks. Moreover, MoMa (Full) surpasses JMP-MT in 16 out of 17 tasks with a large average margin of 24.8%, underscoring the advantage of MoMa in selecting and merging the most relevant knowledge modules.

**Performance of baselines** The best performing baseline is JMP-FT (average rank 2.88), followed by JMP-MT (average rank 3.94). Though additionally trained on upstream tasks of MoMa Hub, JMP-MT is still inferior to JMP-FT. We extrapolate that knowledge conflicts between the disparate material tasks poses tremendous risk to the multi-task learning scheme. We also observe that methods equipped with the JMP encoder achieve better performance than those using CGCNN encoders. This showcases the good transferability of large force field models to material property prediction tasks.

### 3.3 Ablation Study of Adaptive Module Composition

**Setup** We conduct a fine-grained ablation study of the Adaptive Module Fusion algorithm. The following ablated variants are tested: (1) **Select average**, which discards the weights optimized in Eq. (3) and apply arithmetic averaging for the selected modules; (2) **All average**, which simple averages all modules in MoMa Hub; (3) **Random selection**, which picks a random set of modules in MoMa Hub with the same module number as AMC. Further analysis experiments are done using the MoMa's full parametrization, *i.e.*, MoMa (Full), due to its superior performance.

**Results** A visualization of the ablation results on all downstream tasks are shown in Fig. 4. Select average, all average and random selection are inferior to the main results using AMC in 13, 15 and 15 tasks, with an average increase of test MAE of 11.0%, 18.0% and 20.2%. This demonstrates the effectiveness of both the module selection and weighted composition strategies of AMC.

### 3.4 Performance in Few-shot Settings

**Motivation & Setup** To better extrapolate the performance of MoMa in real-world materials discovery scenarios, where candidates with labeled properties are costly to acquire and often exceptionally scarce (Abed et al., 2024), we manually construct a few-shot learning setting and compare the performance of MoMa with JMP-FT, the strongest baseline method. For every downstream task, we randomly down-sample $N$ data points from the train set to construct the few-shot train set, on which we run the AMC algorithm to select modules from MoMa Hub. Then we perform downstream adaptation by fine-tuning on the $N$ data points. The validation and test sets remain consistent with those of the standard settings to ensure a robust evaluation of model performance. Experiments are conducted with $N$ set to 100 and 10, representing few-shot and extremely few-shot scenarios.

**Results** The average test losses for the 17 downstream tasks of MoMa compared to JMP-FT across the full-data, 100-data and 10-data settings are illustrated in Fig. 5. As expected, the test loss increases as the data size decreases, and MoMa consistently outperforms JMP-FT in all settings. Notably, the performance advantage of MoMa is more pronounced in the few-shot settings, with the normalized loss margin widening from 0.03 in full-data setting to 0.11 in 100-data setting and 0.15 in 10-data setting. This suggests that MoMa may offer even greater performance benefits in real-world applications, where the availability of property labels is often limited for effectively fine-tuning large pre-trained models. Complete results are shown in Tab. 4.

### 3.5 Continual Learning Experiments

**Motivation & Setup** Continual learning refers to the ability of an intelligent system to progressively improve by integrating new knowledge (Wang et al., 2024). We explore this capability of MoMa by incorporating new modules into MoMa Hub. Due to its modular nature, it is expected that MoMa will exhibit enhanced performance in tasks that are closely aligned with the new modules, while maintaining its performance when these additions are less relevant. We expand MoMa Hub to include the QM9 dataset (Ramakrishnan et al., 2014) and test the results on all the 17 benchmark material property prediction tasks. For more details on the setup, please refer to Appendix B.4.

**Results** We draw the scatter plot of the reduction rate of test MAE wrt. the proxy error decrease in Fig. 6 across the datasets where QM9 modules are selected. We observe that: (1) The integration of QM9 modules leads to an average of 1.7% decrease in test set MAE; (2) a larger decrease in the AMC optimized proxy error correlates with greater performance improvements post-fine-tuning (with a Pearson correlation of 0.69). We highlight the task of MP Phonons prediction, which marks a significant 11.8% drop in test set MAE after the expansion of MoMa Hub.

### 3.6 Materials Insights Mining

**Motivation** We argue that the AMC weights obtained in Eq. (3) could provide interpretability for MoMa as well as valuable insights into material properties. To explore this, we interpret the weights as indicators for the relationships between MoMa Hub modules and downstream prediction tasks. Following Chang et al. (2022), we present a log-normalized visualization of these weights in Fig. 7.

**Results** We make several interesting observations:

- The weights assigned by AMC effectively captures physically intuitive relationships between material properties. For example, the tasks of experimental band gap (row 1) and experimental formation energy (row 2) assign the highest weights to the computational band gap (column 2 and 14) and formation energy modules (column 1, 12 and 15) in MoMa Hub. Also, for the task of predicting electronic dielectric constants, MoMa's band gap modules are assigned high weights, which is also reasonable given that the dielectric constant is inversely proportional to the square of the band gap (Ravichandran et al., 2016).
- Some less-intuitive relationships have also emerged. For the task of experimental band gap prediction (row 1), the formation energy module from the Materials Project (column 1) is assigned the second-highest weight. For prediction of dielectric constant (row 9), modules related to thermoelectric and thermal properties (column 5 and 6) are non-trivially weighted. However, the

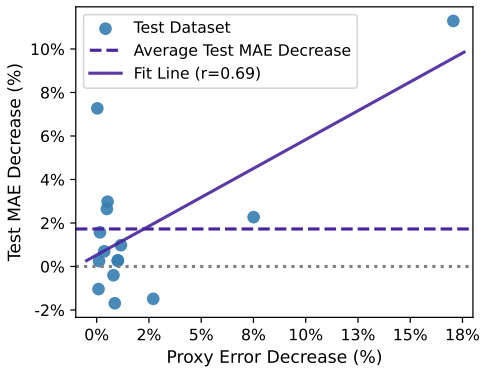


Figure 6: Scatter plot showing the relationship between the test MAE decrease and the proxy error (defined in Eq. (3)) decrease after the addition of QM9 modules. The dashed line represents the average test MAE decrease. The solid line fits the results with linear regression.

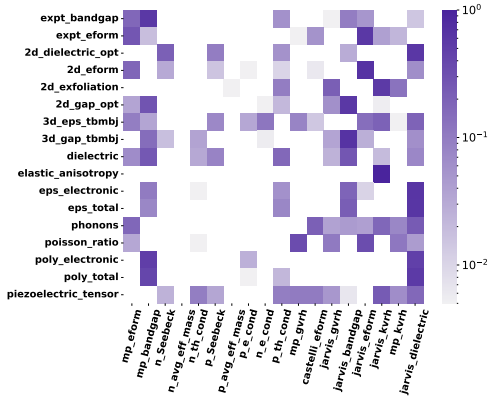


Figure 7: Heat map illustrating the AMC weights on one data split. The x-axis represents the task names of the MoMa Hub modules, while the y-axis shows the 17 material tasks in Tab. 1. Darker colors indicates a stronger correlation between the MoMa module and the downstream tasks.

first-principles relationship between these tasks is indirect. We hypothesize that aside from task relevance, other factors, such as data distribution and data size, may also influence the weight assignments for AMC. Further investigation of these results are left for future work.

## 4 Related Work

**Material Property Prediction with Deep Learning** Deep learning methods have been widely applied for predicting material properties (De Breuck et al., 2021). One series of research (Choudhary & DeCost, 2021; Yan et al., 2022; Das et al., 2023; Lin et al., 2023; Yan et al., 2024; Taniai et al., 2024) have focused on improving neural network architectures to better model the inductive biases of crystals for property prediction tasks, while another line of work develops pre-training strategies on potential energy surface data (Merchant et al., 2023; Batatia et al., 2023; Yang et al., 2024b; Neumann et al., 2024; Barroso-Luque et al., 2024) to facilitate material property prediction. Extending beyond the prevailing pre-train and fine-tune paradigm, MoMa devises effective strategies to centralize material knowledge into modules and adaptively compose the modules to achieve superior downstream performance.

**Modular Deep Learning** Modular deep learning (Pfeiffer et al., 2023) represents a promising paradigm in deep learning, where parameterized modules are composed, selected, and aggregated during the network training process. Examples of modular networks include mixture-of-experts (Jacobs et al., 1991), adapters (Houlsby et al., 2019) and LoRA (Hu et al., 2021). Recently, we have seen an increasing number of successful applications of modular deep learning across domains such as NLP and CV (Puigcerver et al., 2020; Huang et al., 2023; Zhang et al., 2023; Pham et al., 2024), where its strengths in flexibility and minimizing negative interference have been demonstrated. In the field of material property prediction, the idea of modular deep learning is still under-explored, and MoMa marks the first systematic effort to devise modular deep learning framework for materials.

## 5 Conclusion and Outlook

In this paper, we present MoMa, a modular deep learning framework for material property prediction. Motivated by the challenges of diversity and disparity, MoMa first trains specialized modules across a wide spectrum of material tasks, constituting MoMa Hub. We then introduce the Adaptive Module Composition algorithm, which facilitates tailored adaptation from MoMa Hub to each downstream task by adaptively composing synergistic modules. Experimental results across 17

datasets demonstrate the superiority of MoMa, with few-shot and continual learning experiments further highlighting its data-efficiency and scalability.

Finally, we discuss the prospects of MoMa in driving practical advancements in materials discovery. As an open-source platform enabling materials knowledge modularization and distribution, MoMa offers several key advantages: (1) secure, flexible upload of material modules to MoMa Hub without compromising proprietary data; (2) efficient customization of modules for downstream tasks; (3) enhanced property prediction accuracies, even in low-data scenarios. We envision MoMa facilitating a new paradigm of modular material learning and fostering broader community collaboration toward accelerated materials discovery.

## ACKNOWLEDGMENTS

The authors thank Junwei Yang, Mianzhi Pan, Yuanhang Tang, Fanyou Meng for their valuable feedback on the paper. We also thank the anonymous reviewers for reviewing the draft. This work is supported by the National Science and Technology Major Project (2022ZD0117502), the Natural Science Foundation of China (Grant No. 62376133, 62406170), Beijing Nova Program (20240484682) and Wuxi Research Institute of Applied Technologies, Tsinghua University under Grant 20242001120.

## REFERENCES

Jehad Abed, Jiheon Kim, Muhammed Shuaibi, Brook Wander, Boris Duijf, Suhas Mahesh, Hyeonseok Lee, Vahe Gharakhanyan, Sjoerd Hoogland, Erdem Irtem, et al. Open catalyst experiments 2024 (ocx24): Bridging experiments and computational models. *arXiv preprint arXiv:2411.11783*, 2024.

Luis Barroso-Luque, Muhammed Shuaibi, Xiang Fu, Brandon M Wood, Misko Dzamba, Meng Gao, Ammar Rizvi, C Lawrence Zitnick, and Zachary W Ulissi. Open materials 2024 (omat24) inorganic materials dataset and models. *arXiv preprint arXiv:2410.12771*, 2024.

Ilyes Batatia, Philipp Benner, Yuan Chiang, Alin M Elena, Dávid P Kovács, Janosh Riebesell, Xavier R Advincula, Mark Asta, Matthew Avaylon, William J Baldwin, et al. A foundation model for atomistic materials chemistry. *arXiv preprint arXiv:2401.00096*, 2023.

Ivano E Castelli, David D Landis, Kristian S Thygesen, Søren Dahl, Ib Chorkendorff, Thomas F Jaramillo, and Karsten W Jacobsen. New cubic perovskites for one-and two-photon water splitting using the computational materials repository. *Energy & Environmental Science*, 5(10):9034–9043, 2012.

Rees Chang, Yu-Xiong Wang, and Elif Ertekin. Towards overcoming data scarcity in materials science: unifying models and datasets with a mixture of experts framework. *npj Computational Materials*, 8(1):242, 2022.

Kamal Choudhary and Brian DeCost. Atomistic line graph neural network for improved materials property predictions. *npj Computational Materials*, 7(1):185, 2021.

Kamal Choudhary, Irina Kalish, Ryan Beams, and Francesca Tavazza. High-throughput identification and characterization of two-dimensional materials using density functional theory. *Scientific reports*, 7(1):5179, 2017.

Kamal Choudhary, Gowoon Cheon, Evan Reed, and Francesca Tavazza. Elastic properties of bulk and low-dimensional materials using van der waals density functional. *Physical Review B*, 98(1): 014107, 2018.

Kamal Choudhary, Kevin F Garrity, Andrew CE Reid, Brian DeCost, Adam J Biacchi, Angela R Hight Walker, Zachary Trautt, Jason Hattrick-Simpers, A Gilad Kusne, Andrea Centrone, et al. The joint automated repository for various integrated simulations (jarvis) for data-driven materials design. *npj computational materials*, 6(1):173, 2020.

Kishalay Das, Pawan Goyal, Seung-Cheol Lee, Satadeep Bhattacharjee, and Niloy Ganguly. Crysmmnet: multimodal representation for crystal property prediction. In *Uncertainty in Artificial Intelligence*, pp. 507–517. PMLR, 2023.

Pierre-Paul De Breuck, Geoffroy Hautier, and Gian-Marco Rignanese. Materials property prediction for limited datasets enabled by feature selection and joint learning with modnet. *npj computational materials*, 7(1):83, 2021.

Maarten De Jong, Wei Chen, Thomas Angsten, Anubhav Jain, Randy Notestine, Anthony Gamst, Marcel Sluiter, Chaitanya Krishna Ande, Sybrand Van Der Zwaag, Jose J Plata, et al. Charting the complete elastic properties of inorganic crystalline compounds. *Scientific data*, 2(1):1–13, 2015a.

Maarten De Jong, Wei Chen, Henry Geerlings, Mark Asta, and Kristin Aslaug Persson. A database to enable discovery and design of piezoelectric materials. *Scientific data*, 2(1):1–13, 2015b.

Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Steven Diamond, Eric Chu, and Stephen Boyd. CVXPY: A Python-embedded modeling language for convex optimization, version 0.2. `http://cvxpy.org/`, May 2014.

Alexander Dunn, Qi Wang, Alex Ganose, Daniel Dopp, and Anubhav Jain. Benchmarking materials property prediction methods: the matbench test set and automatminer reference algorithm. *npj Computational Materials*, 6(1):138, 2020.

Lenz Fiedler, Karan Shah, Michael Bussmann, and Attila Cangi. Deep dive into machine learning density functional theory for materials science and chemistry. *Physical Review Materials*, 6(4): 040301, 2022.

Johannes Gasteiger, Muhammed Shuaibi, Anuroop Sriram, Stephan Günnemann, Zachary Ward Ulissi, C Lawrence Zitnick, and Abhishek Das. Gemnet-oc: Developing graph neural networks for large and diverse molecular simulation datasets. *Transactions on Machine Learning Research*, 2022.

Sean D Griesemer, Yi Xia, and Chris Wolverton. Accelerating the prediction of stable materials with machine learning. *Nature Computational Science*, 3(11):934–945, 2023.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pp. 2790–2799. PMLR, 2019.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. Lorahub: Efficient cross-task generalization via dynamic lora composition. *arXiv preprint arXiv:2307.13269*, 2023.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*, 2022.

Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Label propagation for deep semi-supervised learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5070–5079, 2019.

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, et al. Commentary: The materials project: A materials genome approach to accelerating materials innovation. *APL materials*, 1(1), 2013.

George Kim, SV Meschel, Philip Nash, and Wei Chen. Experimental formation enthalpies for intermetallic phases and other inorganic compounds. *Scientific data*, 4(1):1–11, 2017.

Wenyi Li, Huan-ang Gao, Mingju Gao, Beiwen Tian, Rong Zhi, and Hao Zhao. Training-free model merging for multi-target domain adaptation. *arXiv preprint arXiv:2407.13771*, 2024.

Yuchao Lin, Keqiang Yan, Youzhi Luo, Yi Liu, Xiaoning Qian, and Shuiwang Ji. Efficient approximations of complete interatomic potentials for crystal property prediction. In *International Conference on Machine Learning*, pp. 21260–21287. PMLR, 2023.

Hassan Masood, Tharmakulasingam Sirojan, Cui Ying Toe, Priyank V Kumar, Yousof Haghshenas, Patrick HL Sit, Rose Amal, Vidhyasaharan Sethu, and Wey Yang Teoh. Enhancing prediction accuracy of physical band gaps in semiconductor materials. *Cell Reports Physical Science*, 4(9), 2023.

Amil Merchant, Simon Batzner, Samuel S Schoenholz, Muratahan Aykol, Gowoon Cheon, and Ekin Dogus Cubuk. Scaling deep learning for materials discovery. *Nature*, 624(7990):80–85, 2023.

Mark Neumann, James Gin, Benjamin Rhodes, Steven Bennett, Zhiyi Li, Hitarth Choubisa, Arthur Hussey, and Jonathan Godwin. Orb: A fast, scalable neural network potential. *arXiv preprint arXiv:2410.22570*, 2024.

Ioannis Petousis, David Mrdjenovich, Eric Ballouz, Miao Liu, Donald Winston, Wei Chen, Tanja Graf, Thomas D Schladt, Kristin A Persson, and Fritz B Prinz. High-throughput screening of inorganic compounds for the discovery of novel dielectric and optical materials. *Scientific data*, 4 (1):1–12, 2017.

Guido Petretto, Shyam Dwaraknath, Henrique PC Miranda, Donald Winston, Matteo Giantomassi, Michiel J Van Setten, Xavier Gonze, Kristin A Persson, Geoffroy Hautier, and Gian-Marco Rignanese. High-throughput density-functional perturbation theory phonons for inorganic materials. *Scientific data*, 5(1):1–12, 2018.

Jonas Pfeiffer, Sebastian Ruder, Ivan Vulić, and Edoardo Maria Ponti. Modular deep learning. *arXiv preprint arXiv:2302.11529*, 2023.

Chau Pham, Piotr Teterwak, Soren Nelson, and Bryan A Plummer. Mixturegrowth: Growing neural networks by recombining learned parameters. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2800–2809, 2024.

Joan Puigcerver, Carlos Riquelme, Basil Mustafa, Cedric Renggli, André Susano Pinto, Sylvain Gelly, Daniel Keysers, and Neil Houlsby. Scalable transfer learning with expert models. *arXiv preprint arXiv:2009.13239*, 2020.

Raghunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole Von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific data*, 1(1):1–7, 2014.

Ram Ravichandran, Alan X Wang, and John F Wager. Solid state dielectric screening versus band gap trends and implications. *Optical materials*, 60:181–187, 2016.

Francesco Ricci, Wei Chen, Umut Aydemir, G Jeffrey Snyder, Gian-Marco Rignanese, Anubhav Jain, and Geoffroy Hautier. An ab initio electronic transport database for inorganic materials. *Scientific data*, 4(1):1–13, 2017.

Janosh Riebesell, Rhys EA Goodall, Anubhav Jain, Philipp Benner, Kristin A Persson, and Alpha A Lee. Matbench discovery–an evaluation framework for machine learning crystal stability prediction. *arXiv preprint arXiv:2308.14920*, 2023.

Lars Ruddigkeit, Ruud Van Deursen, Lorenz C Blum, and Jean-Louis Reymond. Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17. *Journal of chemical information and modeling*, 52(11):2864–2875, 2012.

Soumya Sanyal, Janakiraman Balachandran, Naganand Yadati, Abhishek Kumar, Padmini Rajagopalan, Suchismita Sanyal, and Partha Talukdar. Mt-cgcnn: Integrating crystal graph convolutional neural network with multitask learning for material property prediction. *arXiv preprint arXiv:1811.05660*, 2018.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2016.

Nima Shoghi, Adeesh Kolluru, John R Kitchin, Zachary W Ulissi, C Lawrence Zitnick, and Brandon M Wood. From molecules to materials: Pre-training large generalizable models for atomic property prediction. *arXiv preprint arXiv:2310.16802*, 2023.

Tatsunori Taniai, Ryo Igarashi, Yuta Suzuki, Naoya Chiba, Kotaro Saito, Yoshitaka Ushiku, and Kanta Ono. Crystalformer: infinitely connected attention for periodic structure encoding. *arXiv preprint arXiv:2403.11686*, 2024.

Amanda Wang, Ryan Kingsbury, Matthew McDermott, Matthew Horton, Anubhav Jain, Shyue Ping Ong, Shyam Dwaraknath, and Kristin A Persson. A framework for quantifying uncertainty in dft energy corrections. *Scientific reports*, 11(1):15496, 2021.

Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Logan Ward, Alexander Dunn, Alireza Faghaninia, Nils ER Zimmermann, Saurabh Bajaj, Qi Wang, Joseph Montoya, Jiming Chen, Kyle Bystrom, Maxwell Dylla, et al. Matminer: An open source toolkit for materials data mining. *Computational Materials Science*, 152:60–69, 2018.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*, pp. 23965–23998. PMLR, 2022.

Tian Xie and Jeffrey C Grossman. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Physical review letters*, 120(14):145301, 2018.

Keqiang Yan, Yi Liu, Yuchao Lin, and Shuiwang Ji. Periodic graph transformers for crystal material property prediction. *Advances in Neural Information Processing Systems*, 35:15066–15080, 2022.

Keqiang Yan, Cong Fu, Xiaofeng Qian, Xiaoning Qian, and Shuiwang Ji. Complete and efficient graph transformers for crystal material property prediction. *arXiv preprint arXiv:2403.11857*, 2024.

Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities. *arXiv preprint arXiv:2408.07666*, 2024a.

Han Yang, Chenxi Hu, Yichi Zhou, Xixian Liu, Yu Shi, Jielan Li, Guanzhi Li, Zekun Chen, Shuizhou Chen, Claudio Zeni, et al. Mattersim: A deep learning atomistic model across elements, temperatures and pressures. *arXiv preprint arXiv:2405.04967*, 2024b.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*, 2024.

Jinghan Zhang, Junteng Liu, Junxian He, et al. Composing parameter-efficient modules with arithmetic operation. *Advances in Neural Information Processing Systems*, 36:12589–12610, 2023.

Zhanpeng Zhou, Zijun Chen, Yilan Chen, Bo Zhang, and Junchi Yan. On the emergence of cross-task linearity in pretraining-finetuning paradigm. In *Forty-first International Conference on Machine Learning*, 2024.

Zhi-Hua Zhou. Learnware: on the future of machine learning. *Frontiers Comput. Sci.*, 10(4):589–590, 2016.

Zhi-Hua Zhou, Jianxin Wu, and Wei Tang. Ensembling neural networks: many could be better than all. *Artificial intelligence*, 137(1-2):239–263, 2002.

## A Algorithm for Adaptive Module Composition

The formal description of the Adaptive Module Composition algorithm is included in Algorithm 1.

**Algorithm 1** Adaptive Module Composition

**Input:** MoMa Hub $\mathcal{H} = \{g_1, g_2, \ldots, g_N\}$, Downstream Task $\mathcal{D}$.
**Output:** adaptive module $g_{\mathcal{D}}$ for $\mathcal{D}$.
**for** each module $g_j \in \mathcal{H}$ **do**
    Encode the input materials in the training set of $\mathcal{D}$ using $g_j$ to obtain $\mathcal{X}^j = \{\mathbf{x}_1^j, \mathbf{x}_2^j, \ldots, \mathbf{x}_m^j\}$.
    **for** each sample $\mathbf{x}_i^j \in \mathcal{X}^j$ **do**
        Compute the predicted label $\hat{y_i}^j$ for $\mathbf{x}_i^j$ using $k$NN following Eq. (1).
    **end for**
**end for**
Optimize the module weights $w_j$ using cvxpy to minimize the proxy error defined in Eq. (2), subject to:
$\sum_{j=1}^{N} w_j = 1$ and $w_j \geq 0$. Denote the optimized weights for the $j$-th module as $w_j^*$.
Compose the final adaptive module $g_{\mathcal{D}}$ by weighted averaging the parameters of the MoMa Hub modules:
$g_{\mathcal{D}} = \sum_{j=1}^{N} w_j^* g_j$
**Return:** The composed module $g_{\mathcal{D}}$.

## B Experimental Details

In this section, we provide more experimental details of MoMa regarding the datasets, implementation, baselines and the continual learning setting.

### B.1 Dataset Details

We primarily adopt the dataset setup proposed by Chang et al. (2022). Specifically, we select 35 datasets from Matminer (Ward et al., 2018) for our study, categorizing them into 18 high-resource material tasks, with sample sizes ranging from 10,000 to 132,000 (an average of 35,000 samples), and 17 low-data tasks, with sample sizes ranging from 522 to 8,043 (an average of 2,111 samples).

The high-resource tasks are utilized for training the MoMa Hub modules, as their larger data volumes are likely to encompass a wealth of transferrable material knowledge. A detailed introduction of these MoMa Hub datasets is included in Tab. 2.

The low-data tasks serve as downstream datasets to evaluate the effectiveness of MoMa and its baselines. This setup mimics real-world materials discovery scenarios, where downstream data are often scarce. To ensure robust and reliable comparison results, we exclude two downstream datasets with exceptionally small data sizes (fewer than 20 testing samples) from our experiments, as their limited data could lead to unreliable conclusions. Detailed introduction is included in Tab. 3.

Following Chang et al. (2022), all datasets are split into training, validation, and test sets with a ratio of 7:1.5:1.5. For the downstream low-data tasks, the splitting is performed randomly for 5 times to ensure the stability of evaluation.

### B.2 Implementation Details of MoMa

**Network Architecture** We now introduce the network architecture of MoMa modules. The JMP (Shoghi et al., 2023) backbone is directly taken as the full module parametrization. JMP is pre-trained on $\sim$ 120 million DFT-generated force-field data across large scale datasets on catalyst and small molecules. JMP is a 6-layer GNN model with around 160M parameters which is based on the GemNet-OC architecture (Gasteiger et al., 2022). Note that MoMa is backbone-agnostic. JMP is selected due to its comprehensive strength across a wide range of molecular and crystal tasks, which

Table 2: Datasets for training MoMa Hub modules.

| Datasets | Num | Description |
|---|---|---|
| MP $E_f$ | 132752 | The energy change during the formation of a compound from its elements. Data from Jain et al. (2013). |
| MP $E_g$ | 106113 | The PBE band gaps, calculated using the Perdew-Burke-Ernzerhof (PBE) functional, represent the energy difference between the valence and conduction bands in a material. Data from Jain et al. (2013). |
| MP $G_{VRH}$ | 10987 | VRH-average shear modulus, an approximate value obtained by averaging the shear modulus of polycrystalline materials. Data from Jain et al. (2013). |
| MP $K_{VRH}$ | 10987 | VRH-average bulk modulu, calculated by averaging the Voigt (upper bound) and Reuss (lower bound) bulk moduli. Data from Jain et al. (2013). |
| n-type $\sigma_e$ | 37390 | n-type $\sigma_e$ measures the material's conductivity performance when electrons are the primary charge carriers. Data from Ricci et al. (2017). |
| p-type $\sigma_e$ | 37390 | Similar to n-type $\sigma_e$, with holes as carriers. Data from Ricci et al. (2017). |
| n-type $\kappa_e$ | 37390 | n-type $\kappa_e$ evaluates the efficiency of n-type materials that can conduct both electricity and heat, which is crucial for understanding its performance in thermoelectric applications. Data from Ricci et al. (2017). |
| p-type $\kappa_e$ | 37390 | Similar to n-type $\kappa_e$, with holes as carriers. Data from Ricci et al. (2017). |
| n-type $S$ | 37390 | n-type $S$ denotes the average conductivity eigenvalue, which measures thermoelectric conversion efficiency in the hole-conducting state when electrons act as the primary charge carriers. Data from Ricci et al. (2017). |
| p-type $S$ | 37390 | Similar to n-type $S$, with holes as carriers. Data from Ricci et al. (2017). |
| n-type $\overline{m}_e^*$ | 21037 | n-type $\overline{m}_e^*$ denotes average eigenvalue of conductivity effective mass, which measures the impact of the electron's effective mass on the electrical conductivity. Data from Ricci et al. (2017). |
| p-type $\overline{m}_e^*$ | 20270 | Similar to n-type $\overline{m}_e^*$, with holes as carriers. Data from Ricci et al. (2017). |
| Perovskite $E_f$ | 18928 | Perovskite $E_f$ refers to heat of formation of perovskite, the amount of heat released or absorbed when the perovskite structure is formed from its constituent elements. Data from Castelli et al. (2012). |
| JARVIS $E_f$ | 25923 | Formation energy from the JARVIS dataset (Choudhary et al., 2020). |
| JARVIS dielectric constant (Opt) | 19027 | Dielectric constant measures the material's ability to polarize in response to an electric field in two-dimensional systems. Data from Choudhary et al. (2020). |
| JARVIS $E_g$ | 23455 | PBE band gaps from the JARVIS dataset (Choudhary et al., 2020). |
| JARVIS $G_{VRH}$ | 10855 | VRH-average shear modulus from the JARVIS dataset (Choudhary et al., 2020). |
| JARVIS $K_{VRH}$ | 11028 | VRH-average bulk modulus from the JARVIS dataset (Choudhary et al., 2020). |

allows us to seamlessly conduct the continual learning experiments. We leave the extrapolation of MoMa to other architectures as future work.

For the adapter module, we follow the standard implementation of adapter layers (Houlsby et al., 2019). Specifically, we insert adapter layers between each layer of the JMP backbone. Each layer consists a downward projection to a bottleneck dimension and an upward projection back to the original dimension.

**Hyper-parameters** For the training of JMP backbone, we mainly follow the hyper-parameter configurations in Shoghi et al. (2023), with slight modifications to the learning rate and batch size. During the module training stage of MoMa, we use a batch size of 64 and a learning rate of 5e-4 for 80 epochs. During downstream fine-tuning, we adopt a batch size of 32 and a learning rate of 8e-5. We set the training epoch as 60, with an early stopping patience of 10 epochs to prevent over-fitting. We adopt mean pooling of embedding for all properties since it performs better than sum pooling in certain tasks (e.g. band gap prediction), which is consistent to findings in Shoghi et al. (2023).

For the adapter modules, we employ BERT-style initialization (Devlin, 2018), with the bottleneck dimension set to the half of the input embedding dimension.

For the Adaptive Module Composition (AMC) algorithm, we set the number of nearest neighbors ($K$ in Eq. (1)) to 5. For the optimization problem formulated in Eq. (3), we utilize the CPLEX optimizer from the cvxpy package (Diamond et al., 2014). AMC is applied separately for each random split of the downstream tasks to avoid data leakage.

Table 3: Downstream evaluation datasets.

| Datasets | Num | Description |
|---|---|---|
| Experimental Band Gap (eV) | 2481 | The band gap of a material as measured through physical experiments. Data from Ward et al. (2018). |
| Formation Enthalpy (eV/atom) | 1709 | The energy change for forming a compound from its elements, crucial for defining Gibbs energy of formation. Data from Wang et al. (2021); Kim et al. (2017). |
| 2D Dielectric Constant | 522 | The dielectric constant of 2D materials from Choudhary et al. (2017). |
| 2D Formation Energy (eV/atom) | 633 | The energy change associated with the formation of 2D materials from their constituent elements. Data from Choudhary et al. (2017). |
| Exfoliation Energy (meV/atom) | 636 | The energy required to separate a single or few layers from a bulk materials. Data from Choudhary et al. (2017). |
| 2D Band Gap (eV) | 522 | The band gap of 2D materials from Choudhary et al. (2017). |
| 3D Poly Electronic | 8043 | Poly electronic of 3D materials from Choudhary et al. (2018). |
| 3D Band Gap (eV) | 7348 | The band gap of 3D materials from Choudhary et al. (2018). |
| Refractive Index | 4764 | The quantitative change of the speed of light as it passes through different media. Data from Dunn et al. (2020); Petousis et al. (2017). |
| Elastic Anisotropy | 1181 | The directional dependence of a material's elastic properties. Data from De Jong et al. (2015a). |
| Electronic Dielectric Constant | 1296 | Electronic dielectric constant refers to the dielectric response caused by electronic polarization under an applied electric field. Data from Petretto et al. (2018). |
| Dielectric Constant | 1296 | Dielectric constant of materials from Petretto et al. (2018). |
| Phonons Mode Peak | 1265 | Phonon mode peak refers to the peak in the phonon spectrum caused by specific phonon modes. Data from Petretto et al. (2018). |
| Poisson Ratio | 1181 | Poisson Ratio quantifies the ratio of transverse strain to axial strain in a material under uniaxial stress, reflecting its elastic deformation behavior. Data from De Jong et al. (2015a). |
| Poly Electronic | 1056 | The Average eigenvalue of the dielectric tensor's electronic component, where the dielectric tensor links a material's internal and external fields. Data from Petousis et al. (2017). |
| Poly Total | 1056 | The Average dielectric tensor eigenvalue. Data from Petousis et al. (2017). |
| Piezoelectric Modulus | 941 | Piezoelectric modulus measures a material's ability to convert mechanical stress into electric charge or vice versa. Data from De Jong et al. (2015b). |

**Computational Cost** Experiments are conducted on NVIDIA A100 80 GB GPUs. During the module training stage, training time ranges from 30 to 300 GPU hours, depending on the dataset size. While this training process is computationally expensive, it is a one-time investment, as the trained models are stored in MoMa Hub as reusable material knowledge modules. Downstream fine-tuning requires significantly less compute, ranging from 2 to 8 GPU hours based on dataset scale. The full module and adapter module require similar training time; however, the adapter module greatly reduces memory consumption during training.

### B.3 Baseline Discussion

The CGCNN baseline refers to fine-tuning the CGCNN model (Xie & Grossman, 2018) separately on 17 downstream tasks. Conversely, MoE-(18) involves training individual CGCNN models for each datasets in MoMa Hub and subsequently integrating these models using mixture-of-experts (Jacobs et al., 1991; Shazeer et al., 2016). For the baseline results of CGCNN and MoE-(18), we adopt the open-source codebase provided by Chang et al. (2022) and follows the exactly same parameters as reported in their papers for the result duplication.

For JMP-FT, we use the JMP (large) checkpoint from the codebase open-sourced by Shoghi et al. (2023) and fine-tune it directly on the downstream tasks with a batch size of 64. JMP-MT adopts a multi-task pre-training strategy, training on all 18 MoMa Hub source tasks without addressing the conflicts between disparate material tasks. Starting from the same pre-trained checkpoint as JMP-FT, JMP-MT employs proportional task sampling and trains for 5 epochs across all tasks with a batch size of 16. The convergence of multi-task pre-training is indicated by a lack of further decrease in validation error on most tasks after 5 epochs. For downstream fine-tuning, both JMP-FT and JMP-MT adopt the same training scheme as the fine-tuning stage in MoMa.

Table 4: Test set MAE and average test loss of JMP-FT and MoMa under the full-data, 100-data and 10-data setting. Results are averaged over five random data splits on one random seed. Results are preserved to the third significant digit.

| **Datasets** | **JMP-FT** | **MoMa** | **JMP-FT (100)** | **MoMa (100)** | **JMP-FT (10)** | **MoMa (10)** |
|---|---|---|---|---|---|---|
| Experimental Band Gap | 0.380 | 0.305 | 0.660 | 0.469 | 1.12 | 1.245 |
| Formation Enthalpy | 0.156 | 0.0821 | 0.273 | 0.101 | 0.514 | 0.143 |
| 2D Dielectric Constant | 2.45 | 1.90 | 3.19 | 2.35 | 7.74 | 3.31 |
| 2D Formation Energy | 0.135 | 0.0470 | 0.366 | 0.113 | 0.842 | 0.214 |
| 2D Exfoliation Energy | 38.9 | 36.1 | 54.4 | 56.1 | 118 | 87.3 |
| 2D Band Gap | 0.611 | 0.366 | 0.890 | 0.517 | 1.23 | 1.05 |
| 3D Poly Electronic | 23.7 | 23.0 | 33.6 | 24.8 | 54.0 | 48.9 |
| 3D Band Gap | 0.249 | 0.201 | 1.71 | 0.686 | 2.10 | 1.47 |
| Dielectric Constant | 0.0552 | 0.0535 | 0.134 | 0.102 | 0.289 | 0.231 |
| Elastic Anisotropy | 2.11 | 2.85 | 4.85 | 3.79 | 4.02 | 5.26 |
| Electronic Dielectric Constant | 0.108 | 0.0903 | 0.260 | 0.178 | 0.568 | 0.500 |
| Total Dielectric Constant | 0.172 | 0.155 | 0.361 | 0.287 | 0.543 | 0.527 |
| Phonons Mode Peak | 0.0710 | 0.0521 | 0.221 | 0.199 | 0.493 | 0.485 |
| Poisson Ratio | 0.0221 | 0.0203 | 0.0345 | 0.0317 | 0.0466 | 0.057 |
| Poly Electronic | 2.10 | 2.13 | 3.24 | 2.88 | 6.08 | 5.10 |
| Total Poly | 4.83 | 4.76 | 6.54 | 6.32 | 11.2 | 10.1 |
| Piezoelectric Modulus | 0.169 | 0.175 | 0.248 | 0.258 | 0.303 | 0.290 |
| **Average Test Loss** | 0.222 | 0.187 | 0.408 | 0.299 | 0.700 | 0.550 |

We highlight the two key differences that distinguishes MoMa from MoE-(18): (1) MoE-(18) loads all pre-trained models indiscriminately for each downstream task, whereas MoMa adaptively composes a subset of relevant modules to mitigate knowledge conflicts and encourage positive transfer. (2) MoE-(18) is designed to address the data scarcity issue and is limited to the mixture-of-experts approach, while MoMa introduces modularity to target the inherent challenges in materials science and is not restricted to any specific modular method. Hence, MoMa marks the first systematic effort to devise modular deep learning framework for materials.

### B.4 Details on Continual Learning Experiments

The QM9 dataset (Ramakrishnan et al., 2014) comprises 12 quantum chemical properties (including geometric, electronic, energetic and thermodynamic properties) for 134,000 stable small organic molecules composed of CHONF atoms, drawn from the GDB-17 database (Ruddigkeit et al., 2012). It is widely served as a comprehensive benchmarking dataset for prediction methods of the structure-property relationships in small organic molecules.

In the continual learning experiments, we expand the MoMa hub by including modules trained on the QM9 dataset. For module training, we adopt the same training scheme as the original MoMa modules, with the exception of using sum pooling instead of mean pooling, as it has been empirically shown to perform better (Shoghi et al., 2023).

## C More Experimental Results

We report the complete few-shot learning results in Tab. 4.