TOWARDS EXTRAPOLATION IN DEEP MATERIAL PROPERTY REGRESSION

Mianzhi Pan[1,2] **Jianfei Li**[1] **Yawen Ouyang**[2] **Wei-Ying Ma**[2]
Jianbing Zhang[1,3*] **Hao Zhou**[2*]
[1] National Key Laboratory for Novel Software Technology, Nanjing University, China &
School of Artificial Intelligence, Nanjing University, China;
[2] Institute of AI Industry Research (AIR), Tsinghua University
[3] Chemistry and Biomedicine Innovation Center (ChemBIC), Nanjing University, China
{panmz,lijf}@smail.nju.edu.cn,
{ouyangyawen,zhouhao}@air.tsinghua.edu.cn

ABSTRACT

Deep learning methods have yielded exceptional performances in material property regression (MPR). However, most existing methods operate under the assumption that the training and test are independent and identically distributed (i.i.d.). This overlooks the importance of extrapolation - predicting material properties beyond the range of training data - which is essential for advanced material discovery, as researchers strive to identify materials with exceptional properties that exceed current capabilities. In this paper, we address this gap by introducing a comprehensive benchmark comprising seven tasks specifically designed to evaluate extrapolation in MPR. We critically evaluate existing methods such as deep imbalanced regression (DIR) and regression data augmentation (DA) methods, and reveal their limitations in extrapolation tasks. To address these issues, we propose an incredibly simple Matching-based EXtrapolation (MEX) framework, which reframes MPR as a material-property matching problem to alleviate the inherent complexity of the direct material-to-label mapping paradigm for better extrapolation. Our experimental results show that MEX outperforms all existing methods and demonstrates exceptional capability in identifying promising materials, underscoring its potential for advancing material discovery. Code is available at https://github.com/panmianzhi/Matching-based-EXtrapolation.

1 INTRODUCTION

Material property regression (MPR), the task of predicting continuous material property values, plays a critical role in material discovery across diverse applications such as catalysts and batteries. Traditional *ab initio* calculations, such as Density Functional Theory (DFT), while accurate, are often computationally prohibitive for high-throughput screening. To address this challenge, deep learning models (Xie & Grossman, 2018; Schütt et al., 2021; Yan et al., 2022; Liao et al., 2024; Shoghi et al., 2024) have emerged as efficient alternatives, providing rapid predictions that facilitate the identification of promising material candidates for further validation through detailed simulations or experiments.

Predicting property values outside the scope of all existing materials, known as extrapolation, is a crucial yet overlooked area in deep MPR. A common research problem in materials science is to discover novel materials with higher/lower properties than all known ones, such as organic light-emitting diodes (OLEDs) with extreme color purity (Xu et al., 2020b; Kim & Yasuda, 2022) and semiconductor materials with extraordinary thermodynamic stability (Castelli et al., 2012a;b). Identifying such materials with outstanding properties hinges on effective extrapolation approaches. Previous work (Hatakeyama-Sato & Oyaizu, 2021; Shimakawa et al., 2024; Segal et al., 2024) has studied this problem but relies on sophisticated hand-craft material descriptors, limiting their gener-

*Corresponding author

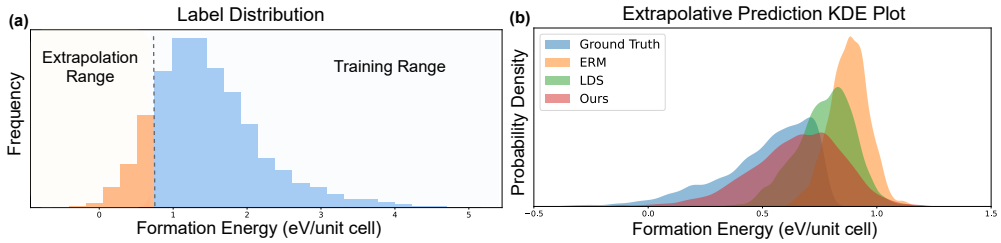

Figure 1: (a) Extrapolation in MPR (for Formation Energy), which aims to generalize to label range outside the training data. (b) Existing methods fail in extrapolative prediction, while our approach predicts properties closer to the real distribution.

alizability when adapted to deep learning-based MPR. As a result, deep extrapolative MPR remains an underexplored area of research, presenting significant opportunities for advancement.

To address this gap, this study conducts a thorough investigation into the extrapolative performance of deep regression methods for predicting diverse material properties, revealing that extrapolating material properties remains a significant challenge. First, a comprehensive benchmark is established using four DFT-calculated datasets from `Matminer` (Ward et al., 2018). To simulate realistic extrapolation scenarios, we divide the whole dataset into train and test sets with disjoint target ranges. An example dataset is given in Figure 1(a). Then, we systematically evaluate existing methods under a wide range of (1) backbones, including representative equivariant geometric GNNs such as PaiNN (Schütt et al., 2021) and EquiformerV2 (Liao et al., 2024); (2) training algorithms, including classic ERM, deep imbalanced regression (DIR) methods (Yang et al., 2021; Gong et al., 2022; Ren et al., 2022; Keramati et al., 2024), data augmentation techniques (Yao et al., 2022; Kaufman & Azencot, 2024); and a general nonlinearity encoding method (Na & Park, 2022). The benchmark outcomes demonstrate a significant degradation in the extrapolative performance of current methods, highlighting the need for more tailored methodologies for this challenge.

In response, we propose **M**atching-based **EX**trapolation (MEX), a novel and remarkably easy framework that reframes MPR as a material-property matching problem, aimed at simplifying the complexity of target functions to enhance model extrapolation. Our motivation is that matching reduces the learning difficulty compared to precise prediction, thereby improving extrapolation. Specifically, MEX employs two complementary training objectives to learn aligned feature spaces for material and property representation matching. First, it performs absolute matching optimization using negative cosine similarity loss, which pulls paired material and label representations closer together. Second, MEX leverages Noise Contrastive Estimation (NCE) (Gutmann & Hyvärinen, 2010) to force the model to distinguish between target and noisy labels. Within the well-aligned latent spaces, MEX predicts by optimizing for the nearest target value for a given sample. Experiments show that MEX not only achieves the best performance on our benchmark (Figure 1(b)) but also exhibits extraordinary detection capability for promising materials, demonstrating superior extrapolation capabilities and potential for more robust material discovery.

Our contributions are summarized as follows:

- **A novel setting of MPR**: We highlight the critical importance of extrapolation in MPR, an area that has been previously understudied yet holds significant implications for realistic material design scenarios.
- **A thorough benchmark for extrapolation in MPR**: We curate a comprehensive benchmark specifically designed to evaluate extrapolation in material properties regression, and thoroughly investigate the effectiveness of diverse techniques such as deep imbalanced regression (DIR) on extrapolation tasks, revealing their limitations in handling the complexities of MPR.
- **An advanced method with exceptional performance**: We propose MEX, a simple yet effective framework that substantially enhances extrapolation capabilities, achieving new state-of-the-art performance across diverse datasets and evaluation metrics.

2 Related Work

2.1 EXTRAPOLATIVE MATERIAL PROPERTY PREDICTION

Extrapolation to unseen material properties has been explored in various studies. Hatakeyama-Sato & Oyaizu (2021) utilized generative models to recover randomly masked input features together with property values. Shimakawa et al. (2024) introduced interpretable quantum-mechanical descriptors combined with interactive linear regression. Segal et al. (2024) applied Bilinear Transduction (Netanyahu et al., 2023) to material property prediction. These approaches depend on heuristic material descriptors, whose limited availability and weak expressiveness constrain their broader applicability. In contrast, deep learning models directly extract features from material structures and have achieved state-of-the-art performance on extensive benchmarks (Dunn et al., 2020; Choudhary et al., 2024). Building on this progress, our work investigates extrapolation in deep material property prediction, focusing on improving generalization to property values beyond the training data.

2.2 DEEP MATERIAL PROPERTY PREDICTION

Recent years have witnessed the tremendous impact of deep learning on predicting material properties (Schütt et al., 2018; Yan et al., 2022; Shoghi et al., 2024). Considering the 3D atomic systems' essence of material structure, numerous studies have aimed to enhance neural architectures to effectively capture the intrinsic physical symmetries of such data. SchNet (Schütt et al., 2018) and CGCNN (Xie & Grossman, 2018) pioneered the use of graph neural networks for 3D atomic systems, which modeled the pairwise atomic distance variant with regard to Euclidean transformations. Since then, a body of research has focused on encoding higher-order geometric invariants (Klicpera et al., 2020; Gasteiger et al., 2021; Yan et al., 2022) and equivariants (Schütt et al., 2021; Passaro & Zitnick, 2023; Liao et al., 2024).

Another area of focus lies in pre-training to learn transferable material representations (Shoghi et al., 2024; Yang et al., 2024; Song et al., 2024). For instance, Shoghi et al. (2024); Yang et al. (2024) pre-trained inter-atomic force field models and show impressive transfer performance to downstream MPR tasks. Song et al. (2024) employed a self-supervised pre-training task via crystal structure reconstruction based on diffusion models. Orthogonal to existing research efforts, our work focuses on the overlooked issue of extrapolation in deep MPR and approaches it from a unique training strategy perspective, which can use any model architecture and pre-trained model as backbones.

2.3 DEEP LEARNING EXTRAPOLATION

Extrapolation in machine learning typically refers to predicting unseen data outside the training distribution. While prior work has examined how deep models perform extrapolation (Xu et al., 2021; Na & Park, 2022; Netanyahu et al., 2023), they primarily focus on extrapolation w.r.t. the covariate distribution. In contrast, this paper addresses extrapolation w.r.t. the label space, where target values are outside the training support.

Several works view extrapolation as a specific deep imbalanced regression (DIR) scenario and tackle this challenge by sample reweighting (Steininger et al., 2021; Yang et al., 2021; Wang & Wang, 2023), feature space regularization (Gong et al., 2022; Keramati et al., 2024; Zhang et al., 2024) and unbiased training objective (Ren et al., 2022). Although effective, DIR algorithms lack tailored strategies to handle disjoint target label intervals, limiting their applicability. This work directly addresses this challenge by introducing a novel training scheme for MPR. The proposed approach reformulates MPR as a material-property matching problem, advancing beyond the conventional end-to-end prediction used in existing DIR methods.

3 METHODOLOGY

3.1 PROBLEM DEFINITION

MPR extrapolation aims to predict unobserved material property values outside the training label range. Formally, let the input space and label space be denoted as $\mathcal{X}$ and $\mathcal{Y}$, where $\mathcal{X}$ contains the structural data of materials, with each sample represented as $\{a_k, p_k\}_{k=1}^{n}$, where a_k and p_k denote the atomic number and position of the k-th atom out of n atoms, respectively. $\mathcal{Y} \subset \mathbb{R}$ corresponds to a continuous range of labels. The training domain and target domain are respectively defined as

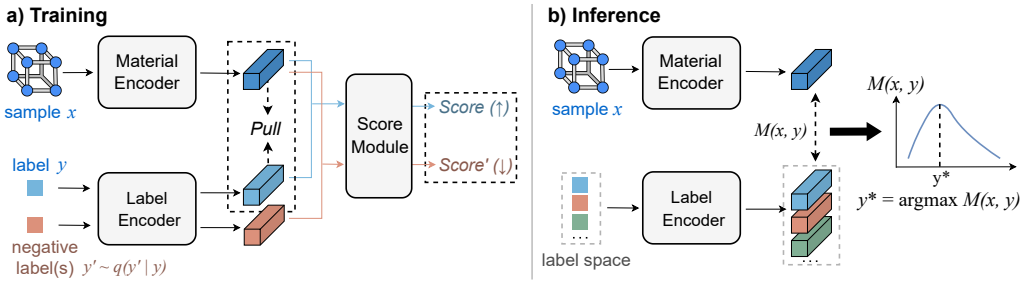

Figure 2: The framework of MEX. (a) During training, MEX begins by drawing negative labels from a mixture Gaussian distribution. Both samples and labels are embedded into the feature space, where the absolute matching optimization aligns the sample with its target label by pulling their representations closer. Noise Contrastive Estimation loss is then applied to refine the feature space by maximizing the score between the sample and its correct label while minimizing the scores between the sample and noisy labels. (b) During testing, MEX predicts the label by identifying the one most similar to the sample in the learned feature space.

$\mathcal{D}_{\text{train}} = \{(x, y) \mid (x, y) \in \mathcal{X} \times \mathcal{Y}_{\text{train}}\}$ and $\mathcal{D}_{\text{target}} = \{(x, y) \mid (x, y) \in \mathcal{X} \times \mathcal{Y}_{\text{target}}\}$, where $\mathcal{Y}_{\text{train}}$ and $\mathcal{Y}_{\text{target}}$ are two disjoint subspaces of $\mathcal{Y}$, i.e.,

$$\mathcal{Y}_{\text{target}} \subset \left\{y \in \mathcal{Y} \mid y > \max(\mathcal{Y}_{\text{train}}) \vee y < \min(\mathcal{Y}_{\text{train}})\right\}$$

Our goal is to develop a model $f : \mathcal{X} \to \mathcal{Y}$ that minimizes the extrapolation error $\mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{target}}}[\ell(f(x), y)]$, where $\ell : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is the loss function. Note that the model can only utilize $\mathcal{D}_{\text{train}}$ without further adapting to $\mathcal{D}_{\text{target}}$ during training.

3.2 Matching-based Extrapolation

Given a training set of N examples $D_{\text{train}} = \{(x_i, y_i)\}_{i=1}^{N}$, the goal of matching-based extrapolation is to learn a binary matching function $\mathcal{M}(x, y) : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ that reflects the matching degrees between materials and properties, i.e., $\mathcal{M}(x, y)$ assigns higher values to paired samples x and labels y, while assigning lower values to unpaired ones. At inference time, MPR (given x) can be reformulated as

$$y^\star = \text{argmax}_{y\in\mathcal{Y}}\mathcal{M}(x, y). \tag{1}$$

Since $\mathcal{Y}$ is a continuous space, we can optimize for $y^\star$ via sampling or optimization algorithms.

In the following, we first outline the motivation behind MEX in Section 3.2.1, followed by the parameterization of the matching function in Section 3.2.2. Next, we detail the training process in Section 3.2.3, where the matching function is optimized from both absolute and relative perspectives. Finally, a stochastic optimization algorithm is introduced in Section 3.2.4 to efficiently solve Equation (1).

3.2.1 Motivation

Compared to directly performing numerical regression, matching-based approach fundamentally reduces the complexity of learning, enabling better generalization. Common numerical regression requires unifying all mapping relationships from "material → property" into a single MLP network. However, the relationship between materials and property values often involves highly complex and nonlinear correspondences, making it challenging for a single global model to effectively capture all local scenarios simultaneously. As a result, numerical regression frequently produces overly smoothed predictions that are confined to the training data distribution, severely limiting its ability to extrapolate beyond the observed property range. The matching-based approach tackles this challenge by reframing the problem as a series of localized matching subproblems, where the model independently evaluates the affinity between material structures and property values. This decomposition allows the model to focus on learning distinct relationships within specific regions of the material-property space, thereby effectively reducing the overall complexity. Since reducing task complexity is closely linked to improved generalization (Xu et al., 2020a), the matching-based approach naturally facilitates more efficient and reliable extrapolation.

3.2.2 PARAMETERIZATION

The matching function $\mathcal{M}(x, y)$ is parameterized as $\mathrm{Sim}(\mathcal{E}_s(x), \mathcal{E}_l(y))$, where $\mathcal{E}_s(\cdot) : \mathcal{X} \to \mathbb{R}^d$ represents the material encoder, $\mathcal{E}_l(\cdot) : \mathbb{R} \to \mathbb{R}^d$ represents the label encoder, and $\mathrm{Sim}(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is non-parametric similarity measurement between two vectors which we employ the cosine similarity. Throughout the paper, the encoded material and label are denoted as $\boldsymbol{z}^s$ and $\boldsymbol{z}^l$, respectively. Variables with subscript i correspond to the i-th training sample.

3.2.3 TRAINING STAGE

In this section, we will introduce two training objectives for learning the sample-label matching relationship.

Absolute matching optimization. To establish a strong alignment between materials and their corresponding property values, we introduce an absolute matching optimization objective. Specifically, we leverage the cosine similarity as a metric to quantify the alignment, ensuring that each paired material representation $\boldsymbol{z}_i^s$ and property representation $\boldsymbol{z}_i^l$ are pulled closer together:

$$\mathcal{L}_{abs,i} = -\frac{\boldsymbol{z}_i^s \cdot \boldsymbol{z}_i^l}{||\boldsymbol{z}_i^s|| \times ||\boldsymbol{z}_i^l||} \qquad \mathcal{L}_{abs} = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_{abs,i}, \tag{2}$$

where $\mathcal{L}_{abs,i}$ is the absolute matching loss of the i-th sample and and $\mathcal{L}_{abs}$ is the total loss of N samples.

Relative matching optimization. Although $\mathcal{L}_{abs}$ enables the model to capture the matching relationship between a sample and its corresponding target value, it falls short in modeling the relative relationships among different property values. This limitation is particularly critical in continuous label regression tasks, where the ability to discern fine-grained distinctions between similar property values is essential for accurate prediction. To achieve this, we introduce Noise Contrastive Estimation (NCE) (Gutmann & Hyvärinen, 2010), a contrastive learning objective that enhances the model's capacity to differentiate between target and noisy labels:

$$\mathcal{L}_{nce,i} = -\log \frac{\exp\left\{\mathcal{S}\left(\boldsymbol{z}_i^s, \boldsymbol{z}_{(i,0)}^l\right) - \log q\left(y_{(i,0)} \mid y_i\right)\right\}}{\sum_{m=0}^{M} \exp\left\{\mathcal{S}\left(\boldsymbol{z}_i^s, \boldsymbol{z}_{(i,m)}^l\right) - \log q\left(y_{(i,m)} \mid y_i\right)\right\}}, \tag{3}$$

where $y_{(i,0)}$ indicates the positive value with the representation $\boldsymbol{z}_{(i,0)}^l$, $y_{(i,m)}$ indicates the negative value with the representation $\boldsymbol{z}_{(i,m)}^l$, and $\mathcal{S}(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ refers to a non-linear score module that computes the compatibility score between a sample representation and a label representation.

Constructing effective negative label values poses a significant challenge due to the inherently infinite nature of potential negatives—any value other than y_i could theoretically serve as a negative. To address this, we adopt a strategy that prioritizes hard negatives, which are values in close proximity to the positive label. Specifically, the negative labels are sampled from a mixture of K Gaussians centered around y_i:

$$q(y|y_i) = \frac{1}{K}\sum_{k=1}^{K} \mathcal{N}(y; y_i, \sigma_k^2). \tag{4}$$

The final NCE loss of the training samples is

$$\mathcal{L}_{nce} = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_{nce,i}. \tag{5}$$

By combining the absolute and relative matching optimization, the total training objective is:

$$\mathcal{L} = \mathcal{L}_{nce} + \lambda \mathcal{L}_{abs}, \tag{6}$$

where λ is a trade-off parameter.

Algorithm 1 Inference by stochastic optimization

Input: x: Input sample, $\mathcal{M}$: Matching function
Parameter: C: Number of candidate labels, T: Iterations, l: Label lower bound, u: Label upper bound, σ: initial noise scale, β: noise shrink factor
Output: $y^\star$: Optimal label

1: initialize $\{y_i\}_{i=1}^C, y_i \sim \mathcal{U}([l,u])$ //uniform sample from $[l,u]$
2: **for** $t \leftarrow 1$ **to** T **do**
3: $\{p_i\}_{i=1}^C \leftarrow \text{softmax}(\{\mathcal{M}(x,y_i)\}_{i=1}^C)$ //get each label's probability
4: $\{y_i\}_{i=1}^C \leftarrow \text{sample}(\{y_i\}_{i=1}^C, \{p_i\}_{i=1}^C)$ //sample with replacement
5: **for** $i \leftarrow 1$ **to** C **do**
6: $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$
7: $y_i \leftarrow y_i + \epsilon_i$ //random perturb y_i
8: $y_i \leftarrow \text{clip}(y_i, l, u)$ //clip y_i within $[l,u]$
9: **end for**
10: $\sigma \leftarrow \beta * \sigma$ //noise shrink
11: **end for**
12: $y^\star \leftarrow \text{argmax}_{y \in \{y_i\}_{i=1}^C} \mathcal{M}(x,y)$
13: **return** $y^\star$

3.2.4 Inference Stage

Fast prediction of material properties is crucial, particularly in high-throughput screening. Therefore, an efficient algorithm for solving Equation (1) is essential. To this end, we propose to use a derivative-free stochastic optimization method based on Monte Carlo sampling (Homem-de Mello & Bayraksan, 2014). The complete algorithm is described in Algorithm 1. During inference, C candidate labels are initially sampled within the regression bound and refined iteratively (line 2-10 of Algorithm 1). At each iteration, a probability distribution over the candidate labels is computed proportionally to their matching values with the input (line 3). Based on this distribution, candidates are sampled with replacement (line 4). To further explore the solution space, Gaussian noise with a shrinking scale is added to each label during refinement (lines 6–10). This iterative refinement allows the candidates to converge toward the optimal labels. Finally, the optimal label is selected as the candidate label yielding the highest matching value to the input (line 12). An efficiency analysis of Algorithm 1 is presented in Section 4.5.

4 Benchmarking Extrapolative MPR

4.1 Dataset and Evaluation Metrics

Dataset. Our benchmark employs four datasets from `Matminer` (Ward et al., 2018), covering the following properties: Formation Energy, Shear Modulus, Refractive Index, and Phonons Mode Peak. All these datasets consist of DFT-calculated data, with data points ranging from 1,265 to 18,928. We provide full label distributions and more dataset characteristics such as atom number and lattice constants in Supplementary Figure 5 and table 3.

In order to evaluate the extrapolative performance of MPR models, we partition the training and test sets into disjoint label intervals. Specifically, we divide the datasets so that the test set consists of the 15% of materials with the highest/lowest property values. In addition, we followed the *forward-holdout validation* method proposed by Shimakawa et al. (2024) to define the validation set, comprising the next 15% of materials with the second-highest/lowest property values. Whether the extrapolative side is higher or lower depends on the specific requirement of material design scenarios. For example, researchers aim to identify the structure with the lowest Formation Energy for a given chemical composition as it represents the thermodynamically most stable phase. Thus, for such properties, the extrapolative side is low. For properties where both low and high values are of interest, e.g., Shear Modulus, the dataset is split with both configurations once each to ensure comprehensive evaluation across the spectrum. Train/val/test label range of the resulting seven benchmark datasets are shown in Supplementary Table 4.

Table 1: Test MAE(↓) on the benchmark dataset where BalancedMSE is abbreviated to BMSE. Bold is for the best and italics is for the second best in each column for both models. We report the standard deviation among 3 runs, consistent across all subsequent tables.

Model	Algo	Formation Energy	Shear Modulus		Refractive Index		Phonons Mode Peak		Avg Rank
		low	low	high	low	high	low	high	
PaiNN	ERM	0.424(0.001)	0.613(0.089)	0.363(0)	0.275(0.007)	0.781(0.017)	0.82(0.005)	0.975(0.022)	7.4
	LDS	0.372(0.018)	0.524(0.001)	0.335(0.004)	0.264(0.004)	0.781(0.011)	0.844(0.029)	1.04(0.091)	5.9
	Ranksim	0.42(0.003)	0.54(0.001)	*0.246(0.03)*	0.267(0.004)	0.775(0.002)	*0.732(0.106)*	0.983(0.046)	5
	BMSE	*0.36(0.05)*	*0.462(0.041)*	**0.214(0.038)**	0.269(0.059)	*0.671(0.01)*	0.758(0.011)	1.02(0.018)	*3.4*
	ConR	0.403(0.012)	0.535(0.004)	0.329(0.002)	0.303(0.129)	0.74(0.06)	0.772(0.15)	*0.939(0.007)*	5
	C-Mixup	0.387(0.005)	0.537(0.002)	0.349(0.001)	0.257(0.005)	0.788(0.004)	0.82(0.008)	0.966(0.013)	5.7
	FOMA	0.401(0.004)	**0.446(0.056)**	0.312(0.071)	*0.219(0.025)*	0.746(0.02)	0.776(0.006)	0.991(0.024)	3.9
	ANE	0.545(0.004)	0.528(0.002)	0.349(0.003)	0.256(0.001)	0.8(0.019)	0.843(0.033)	1.041(0.037)	7.1
	MEX	**0.336(0.024)**	0.475(0.003)	0.263(0.01)	**0.148(0.008)**	**0.533(0.005)**	**0.643(0.034)**	**0.896(0.026)**	**1.6**
EquiformerV2	ERM	0.367(0.003)	0.512(0.001)	0.306(0.001)	0.218(0.001)	0.639(0.002)	0.73(0.006)	0.923(0.01)	6.3
	LDS	*0.278(0.008)*	0.4944(0.004)	0.295(0.005)	0.195(0.009)	0.643(0.002)	0.749(0.001)	0.905(0.012)	4.3
	Ranksim	0.356(0.003)	0.467(0.075)	0.304(0.003)	0.217(0.002)	0.624(0.002)	0.727(0.005)	0.915(0.005)	4.4
	BMSE	0.398(0.136)	**0.388(0.028)**	**0.167(0.02)**	*0.184(0.006)*	*0.599(0.013)*	*0.568(0.039)*	1.02(0.022)	*3.4*
	ConR	0.319(0.003)	0.509(0.004)	0.326(0.006)	0.222(0.004)	0.621(0.006)	0.735(0.004)	*0.897(0.009)*	5.1
	C-Mixup	0.312(0.016)	0.509(0.001)	0.314(0.002)	0.205(0.003)	0.626(0.003)	0.752(0.002)	0.915(0.005)	5.6
	FOMA	0.314(0.004)	0.511(0.001)	0.311(0.001)	0.196(0.004)	0.627(0.002)	0.741(0.007)	0.914(0.001)	5.3
	ANE	0.491(0.02)	0.524(0.001)	0.335(0.002)	0.228(0.005)	0.711(0.017)	0.912(0.041)	1.018(0.019)	8.7
	MEX	**0.184(0.012)**	*0.404(0.012)*	*0.235(0.015)*	**0.129(0.001)**	**0.521(0.023)**	**0.502(0.015)**	**0.809(0.018)**	**1.3**

Metrics. We use three regression metrics, including Mean Average Error (MAE), Spearman correlation, and error Geometric Mean (GM). The first two are common metrics for regression and the third is proposed by Yang et al. (2021), and is defined as $\left(\Pi_i^n e_i\right)^{1/n}$, where e_i is the prediction error of the i-th sample.

4.2 Benchmark Methods

Backbones. We employ Geometric Graph Neural Networks (GNNs) (Han et al., 2024), which are designed to process data with geometric structures and have been widely used in material property prediction. Our training framework is architecture-agnostic, and we selected two representative equivariant Geometric GNNs: the GNN-based PaiNN (Schütt et al., 2021) and the Transformer-based EquiformerV2 (Liao et al., 2024). We utilize models implemented by `fairchem`[1].

Algorithms. In the benchmark study, we explore three categories of deep extrapolative methods. The first category is DIR methods. The second is the regression data augmentation (DA) and the third category is the nonlinearity encoding method. To provide a comprehensive evaluation, we assess the performance of several representative methods from each category. Specifically, we choose LDS (Yang et al., 2021), Ranksim (Gong et al., 2022), BalancedMSE (Ren et al., 2022), and Conr (Keramati et al., 2024) for DIR methods; C-Mixup (Yao et al., 2022) and FOMA (Kaufman & Azencot, 2024) for regression DA. Encoding nonlinearity into inputs has been demonstrated to be an effective way for effective covariate extrapolation (Xu et al., 2021; Na & Park, 2022), we consider ANE (Na & Park, 2022) since it is a data-agnostic nonlinearity encoding method which can be seamlessly applied to MPR. All these methods are benchmarked against the empirical risk minimization (ERM) baseline to evaluate their performance.

Implementation details. MEX is a general training framework agnostic to the material encoder. For the label encoder of MEX, we employ a linear layer attached by an activation function. The score module is a 4-layer Multi-layer Perceptron (MLP) that projects the concatenated sample and label representation to a score scalar. Besides, we empirically investigate various implementations of and compare their performance in Section 4.5.

In the training phase, 500 noisy labels are sampled for each example. We simply follow Gustafsson et al. (2020) to set $K = 3$ and $\sigma_1 = 0.075, \sigma_2 = 0.15, \sigma_3 = 0.3$ for the noisy distribution. During inference, the candidate label size is established at $C = 2000$, which is initially sampled uniformly from $[-10, 10]$, which can encompass the theoretical range for most materials-related

[1] `https://github.com/FAIR-Chem/fairchem?tab=readme-ov-file`

properties. Note that the sampling interval can be freely adjusted based on prior knowledge of material properties. The candidate labels are updated for 5 iterations before we make the final prediction. Hyperparameters selection and additional training details are provided in the supplementary.

4.3 Main Results

We report the performance for all methods in Table 1 and Supplementary Tables 5 and 6.

MEX achieves superior extrapolation performance. Under the MAE metric (Table 1), MEX achieves the best average rank across both models, with ranks of 1.6 and 1.3 for EquiformerV2 and PaiNN, respectively. Specifically, MEX attains the lowest MAE on 5 out of 7 datasets for both models. For the remaining two datasets (Shear Modulus-low&high), MEX performs competitively with the best-performing method. Under the GM metric (Supplementary Table 5), MEX continues to outperform all baselines, demonstrating its superior extrapolative ability over existing techniques.

DIR methods are strong baselines for extrapolation. We observe that all DIR methods rank better than ERM on average for both models. For each dataset, at least one DIR method outperforms ERM, demonstrating their effectiveness in extrapolation. Notably, among the DIR methods, BalancedMSE consistently achieves the highest ranking. On certain datasets, it performs competitively to MEX, highlighting its strong extrapolation capability. We recommend that future evaluations on extrapolation consistently include DIR methods, particularly BalancedMSE, as baselines due to their overall robustness.

Data augmentation slightly helps extrapolation, while ANE fails to extrapolate effectively in most cases. Both C-Mixup and FOMA achieve higher overall rankings than ERM, with FOMA only underperforming MEX and BalancedMSE on the PaiNN model. However, it is noteworthy that these methods do not consistently outperform ERM across all datasets, and both exhibit performance comparable to ERM when applied with EquiformerV2. For ANE, it significantly underperforms other baselines. We hypothesize that this is due to ANE's design for covariate extrapolation solely and its strategy of encoding nonlinearity beforehand is unsuitable for predicting unseen target values.

Extrapolation remains a challenging problem. As shown in Table 1, the actual MAEs are relatively large compared to the target values. Furthermore, we observe that all algorithms' predictions exhibit weak (0-0.4) or even negative Spearman correlations with the targets across most datasets (Supplementary Table 6). The Formation Energy dataset, however, is an exception. Most methods achieve stronger correlations on this task, which we hypothesize is due to the availability of sufficient data and the relative simplicity of the material structure in this task, where the materials are perovskite with a general chemical formula ABX_3. In conclusion, accurately predicting extrapolative samples remains a significant challenge for current methods.

4.4 Potential Impact on Cutting-edge Material Discovery

As discussed in Section 4.3, extrapolation presents a significant challenge for methods in the literature, and our approach is no exception. Given such limitations, one may wonder: *To what extent can current deep learning techniques assist in the discovery of cutting-edge materials?*

In addition to accurately predicting the property values of extrapolative samples, we contend that the ability to **detect** materials with potentially groundbreaking properties is also crucial. Once identified, their properties can be further validated by first-principles calculations or wet experiments. Consequently, models' detection capabilities could become vital tools in advancing material discovery.

To assess the extrapolative material detection ability of different methods, we employ a classification metric called *exploration accuracy* (E_{acc}) (Xiong et al., 2020) to evaluate the extraordinary material detection capability of existing methods, defined as $E_{acc} = \frac{\#Positive}{\#Positive+\#Negative}$, where a test sample is marked as positive if its predicted value is outside the training label range, or it is marked as negative. The results are shown in Figure 3. MEX outperforms previous methods in 5 out of 7 detection tasks. Notably, it achieves a recall rate of over 80% on two datasets and exceeds 60% on all datasets. This substantial performance advantage demonstrates the robustness of MEX and highlights its potential to identify cutting-edge materials that might otherwise remain undiscovered.

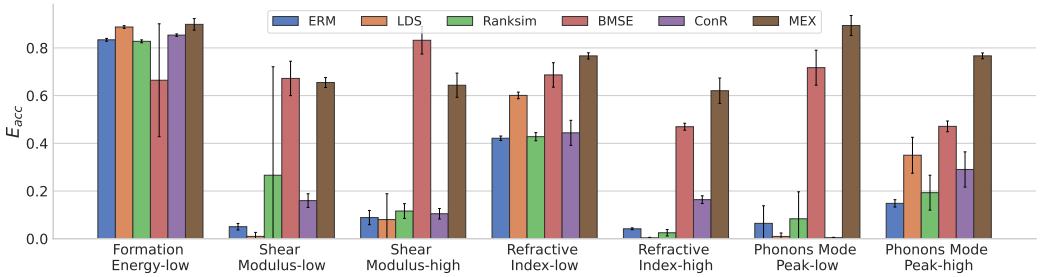

Figure 3: Exploration accuracy of MEX and five DIR methods in detecting extrapolative samples.

4.5 Discussion

Score module analysis. The score module $\mathcal{S}(\cdot,\cdot)$ is a critical component in learning relative matching relationships between sample and label. Here, we investigate the effects of various design choices. The first, referred to as MEX (mlp+cos), employs two independent 2-layer MLPs to project the sample and label representations into a new space, after which the cosine similarity between the two projections is computed. The second approach, MEX (cos), directly computes the cosine similarity between the original sample and label representations.

Table 2: Test MAE of EquiformerV2 on Formation Energy, Refeactive Index(bottom& top) datasets. MEX (cos) and MEX (mlp+cos) denote different designs of the score module in our framework.

Dataset	Formation Energy	Refractive Index	
	low	low	high
MEX (cos)	0.382(0.004)	0.231(0.003)	0.625(0.007)
MEX (mlp+cos)	0.188(0.025)	0.191(0.017)	0.577(0.014)
MEX	**0.184(0.012)**	**0.129(0.001)**	**0.521(0.023)**

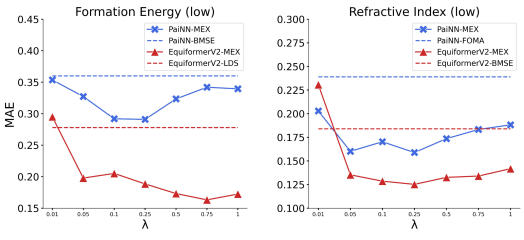

Figure 4: Ablation study on λ.

As illustrated in Table 2, MEX and MEX (mlp+cos) exhibit comparable performance, while MEX (cos) demonstrates inferior performance relative to the other designs. This observation aligns with findings in SimCLR (Chen et al., 2020), which indicate that incorporating a learnable nonlinear transformation on the representations before applying the contrastive loss, rather than directly optimizing the representations, could significantly enhance the quality of the learned features.

Trade-off parameter analysis. We examine the selection of the trade-off parameter λ by assessing model performance across various values of λ. Figure 4 illustrates the performance of MEX alongside prior top-performing methods on three benchmark datasets. As λ changes, MEX consistently surpasses previous approaches across both models, thereby confirming its robustness to diverse hyperparameter configurations and backbone architecture choices.

Running time analysis. Our method requires an iterative refinement of candidate labels before making the final prediction for each testing sample, which inherently results in a longer processing time compared to traditional regression methods. Specifically, this involves encoding 1,500 labels and computing their matching value over 10 iterations during our experiment. Despite this complexity, our experimental results indicate that the average computation time for MEX per test sample is about 0.006s on the NVIDIA 3090, which is comparable to baseline methods (around 0.002s). Thus, the computational overhead associated with our approach remains acceptable.

5 Conclusion

In this work, we shed light on the challenging task of extrapolation in material property regression (MPR), which aims to generalize to materials with unseen property values. We introduce a new benchmark consisting of seven MPR tasks and provide a comprehensive evaluation of existing

methods' extrapolation capabilities. To address the task, we propose a simple yet effective framework that captures the sample-label matching relationship in the latent space. Extensive experiments demonstrate the superior performance of our approach and highlight its potential application in the discovery of cutting-edge materials.

6 ACKNOWLEDGMENTS

This work was supported by the National Science and Technology Major Project (2022ZD0117502), the National Natural Science Foundation of China (Grants No. 62376133 and 62406170), the Beijing Nova Program (20240484682) and the Wuxi Research Institute of Applied Technologies, Tsinghua University (Grant No. 20242001120). We also acknowledge the support from the AI & AI for Science Project of Nanjing University.

REFERENCES

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, 2012a.

Ivano E Castelli, Thomas Olsen, Soumendu Datta, David D Landis, Søren Dahl, Kristian S Thygesen, and Karsten W Jacobsen. Computational screening of perovskite metal oxides for optimal solar light capture. *Energy & Environmental Science*, 5(2):5814–5819, 2012b.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1597–1607. PMLR, 2020. URL http://proceedings.mlr.press/v119/chen20j.html.

Kamal Choudhary, Daniel Wines, Kangming Li, Kevin F Garrity, Vishu Gupta, Aldo H Romero, Jaron T Krogel, Kayahan Saritas, Addis Fuhr, Panchapakesan Ganesh, et al. Jarvis-leaderboard: a large scale benchmark of materials design methods. *npj Computational Materials*, 10(1):93, 2024.

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.

Johannes Gasteiger, Florian Becker, and Stephan Günnemann. Gemnet: Universal directional graph neural networks for molecules. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 6790–6802, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/35cf8659cfcb13224cbd47863a34fc58-Abstract.html.

Yu Gong, Greg Mori, and Frederick Tung. Ranksim: Ranking similarity regularization for deep imbalanced regression. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 7634–7649. PMLR, 2022. URL https://proceedings.mlr.press/v162/gong22a.html.

Fredrik Gustafsson, Martin Danelljan, Radu Timofte, and Thomas B. Schön. How to train your energy-based model for regression. In *31st British Machine Vision Conference 2020, BMVC 2020, Virtual Event, UK, September 7-10, 2020*. BMVA Press, 2020. URL https://www.bmvc2020-conference.com/assets/papers/0154.pdf.

Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In Yee Whye Teh and D. Mike Titterington (eds.), *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, volume 9 of *JMLR Proceedings*, pp. 297–304. JMLR.org, 2010. URL http://proceedings.mlr.press/v9/gutmann10a.html.

Jiaqi Han, Jiacheng Cen, Liming Wu, Zongzhao Li, Xiangzhe Kong, Rui Jiao, Ziyang Yu, Tingyang Xu, Fandi Wu, Zihe Wang, Hongteng Xu, Zhewei Wei, Yang Liu, Yu Rong, and Wenbing Huang. A survey of geometric graph neural networks: Data structures, models and applications. *CoRR*, abs/2403.00485, 2024. doi: 10.48550/ARXIV.2403.00485. URL https://doi.org/10.48550/arXiv.2403.00485.

Kan Hatakeyama-Sato and Kenichi Oyaizu. Generative models for extrapolation prediction in materials informatics. *ACS omega*, 6(22):14566–14574, 2021.

Tito Homem-de Mello and Güzin Bayraksan. Monte carlo sampling-based methods for stochastic optimization. *Surveys in Operations Research and Management Science*, 19(1):56–85, 2014.

A Jain, SP Ong, G Hautier, W Chen, WD Richards, S Dacek, S Cholia, D Gunter, D Skinner, G Ceder, et al. The materials project: a materials genome approach to accelerating materials innovation. apl mater 1: 011002, 2013.

Ilya Kaufman and Omri Azencot. First-order manifold data augmentation for regression learning. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=geajNKab7g.

Mahsa Keramati, Lili Meng, and R. David Evans. Conr: Contrastive regularizer for deep imbalanced regression. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=RIuevDSK5V.

Hyung Jong Kim and Takuma Yasuda. Narrowband emissive thermally activated delayed fluorescence materials. *Advanced Optical Materials*, 10(22):2201714, 2022.

Johannes Klicpera, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=B1eWbxStPH.

Yi-Lun Liao, Brandon M. Wood, Abhishek Das, and Tess E. Smidt. Equiformerv2: Improved equivariant transformer for scaling to higher-degree representations. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=mCOBKZmrzD.

Gyoung S Na and Chanyoung Park. Nonlinearity encoding for extrapolation of neural networks. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1284–1294, 2022.

Aviv Netanyahu, Abhishek Gupta, Max Simchowitz, Kaiqing Zhang, and Pulkit Agrawal. Learning to extrapolate: A transductive approach. *arXiv preprint arXiv:2304.14329*, 2023.

Saro Passaro and C. Lawrence Zitnick. Reducing SO(3) convolutions to SO(2) for efficient equivariant gnns. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 27420–27438. PMLR, 2023. URL https://proceedings.mlr.press/v202/passaro23a.html.

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.

Jiawei Ren, Mingyuan Zhang, Cunjun Yu, and Ziwei Liu. Balanced MSE for imbalanced visual regression. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 7916–7925. IEEE, 2022. doi: 10.1109/CVPR52688.2022.00777. URL https://doi.org/10.1109/CVPR52688.2022.00777.

Kristof Schütt, Oliver T. Unke, and Michael Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 9377–9388. PMLR, 2021. URL http://proceedings.mlr.press/v139/schutt21a.html.

Kristof T Schütt, Huziel E Sauceda, P-J Kindermans, Alexandre Tkatchenko, and K-R Müller. Schnet–a deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, 148(24), 2018.

Nofit Segal, Aviv Netanyahu, Kevin P Greenman, Pulkit Agrawal, and Rafael Gomez-Bombarelli. Known unknowns: Out-of-distribution property prediction in materials and molecules. In *AI for Accelerated Materials Design-NeurIPS 2024*, 2024.

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

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. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=PfPnugdxup.

Zixing Song, Ziqiao Meng, and Irwin King. A diffusion-based pre-training framework for crystal property prediction. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan (eds.), *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pp. 8993–9001. AAAI Press, 2024. doi: 10.1609/AAAI.V38I8.28748. URL https://doi.org/10.1609/aaai.v38i8.28748.

Michael Steininger, Konstantin Kobs, Padraig Davidson, Anna Krause, and Andreas Hotho. Density-based weighting for imbalanced regression. *Mach. Learn.*, 110(8):2187–2211, 2021. doi: 10.1007/S10994-021-06023-5. URL https://doi.org/10.1007/s10994-021-06023-5.

Ziyan Wang and Hao Wang. Variational imbalanced regression: Fair uncertainty quantification via probabilistic smoothing. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/612a56f193d031687683445cd0001083-Abstract-Conference.html.

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.

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.

Zheng Xiong, Yuxin Cui, Zhonghao Liu, Yong Zhao, Ming Hu, and Jianjun Hu. Evaluating explorative prediction power of machine learning algorithms for materials discovery using k-fold forward cross-validation. *Computational Materials Science*, 171:109203, 2020.

Keyulu Xu, Jingling Li, Mozhi Zhang, Simon S. Du, Ken ichi Kawarabayashi, and Stefanie Jegelka. What can neural networks reason about? In *International Conference on Learning Representations*, 2020a. URL https://openreview.net/forum?id=rJxbJeHFPS.

Keyulu Xu, Mozhi Zhang, Jingling Li, Simon Shaolei Du, Ken-ichi Kawarabayashi, and Stefanie Jegelka. How neural networks extrapolate: From feedforward to graph neural networks. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=UH-cmocLJC.

Yincai Xu, Zong Cheng, Zhiqiang Li, Baoyan Liang, Jiaxuan Wang, Jinbei Wei, Zuolun Zhang, and Yue Wang. Molecular-structure and device-configuration optimizations toward highly efficient green electroluminescence with narrowband emission and high color purity. *Advanced Optical Materials*, 8(9):1902142, 2020b.

Keqiang Yan, Yi Liu, Yuchao Lin, and Shuiwang Ji. Periodic graph transformers for crystal material property prediction. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/6145c70a4a4bf353a31ac5496a72a72d-Abstract-Conference.html.

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*, 2024.

Yuzhe Yang, Kaiwen Zha, Ying-Cong Chen, Hao Wang, and Dina Katabi. Delving into deep imbalanced regression. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 11842–11851. PMLR, 2021. URL http://proceedings.mlr.press/v139/yang21m.html.

Huaxiu Yao, Yiping Wang, Linjun Zhang, James Y. Zou, and Chelsea Finn. C-mixup: Improving generalization in regression. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/1626be0ab7f3d7b3c639fbfd5951bc40-Abstract-Conference.html.

Shihao Zhang, Kenji Kawaguchi, and Angela Yao. Deep regression representation learning with topology. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=HbdeEGVfEN.

A DATASET DETAILS

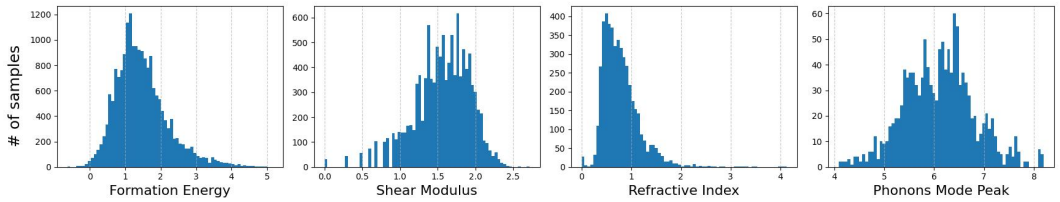

Figure 5: Overview of the label distribution for the original datasets. The X-axis denotes the respective property values. They were divided into seven benchmark datasets.

Table 3: Dataset characteristics, including sample number, atom number (mean and std.), and lattice constants (mean and std.). The symbols $\vec{a}$, $\vec{b}$, and $\vec{c}$ denote the unit cell vectors. The notation $\|\cdot\|$ denotes the length of a vector and $\angle(\cdot,\cdot)$ denotes the angle between two vectors.

Property	**#Samples**	**#Atoms**	$\|\vec{a}\|$	$\|\vec{b}\|$	$\|\vec{c}\|$	$\angle(\vec{b},\vec{c})$	$\angle(\vec{a},\vec{c})$	$\angle(\vec{a},\vec{b})$
Formation Energy Castelli et al. (2012a)	18982	5 ± 0	4.14 (0.31)	4.14 (0.31)	4.14 (0.31)	90.0 (0)	90.0 (0)	90.0 (0)
Shear Modulus Jain et al. (2013)	10987	8.63 ± 8.66	4.96 (1.5)	5.33 (1.67)	6.41 (2.98)	83.29 (20.3)	82.86 (19.78)	85.35 (23.49)
Refractive Index Petousis et al. (2017)	4764	16.9 ± 14.67	5.98 (1.94)	6.6 (2.31)	7.98 (3.61)	86.32 (19.39)	87.07 (19.12)	89.55 (22.47)
Phonons Mode Peak Petretto et al. (2018)	1265	7.53 ± 3.74	5.32 (1.42)	5.66 (1.57)	6.72 (2.09)	83.55 (23.85)	82.95 (23.4)	84.1 (25.15)

Table 4: Details of our curated benchmark datasets.

Property	**Extrapolate side**	**Train label range**	**Val label range**	**Test label range**
Formation Energy	low	[1.06, 5.16]	[0.76, 1.06]	[-0.64, 0.76]
Shear Modulus	low	[1.4, 2.72]	[1.18, 1.4]	[0, 1.18]
	high	[0, 1.78]	[1.78, 1.93]	[1.93, 2.72]
Refractive Index	low	[0.56, 4.13]	[0.45, 0.56]	[0, 0.45]
	high	[0, 0.9]	[0.9, 1.11]	[1.11, 4.13]
Phonons Mode Peak	low	[5.72, 8.2]	[5.41, 5.72]	[4.09, 5.41]
	high	[4.09, 6.45]	[6.45, 6.79]	[6.8, 8.2]

B EXPERIMENT DETAILS

B.1 EVALUATION METRICS

MAE. Mean Absolute Error (MAE) is defined as $\frac{1}{N}\sum_{i=0}^{N}|y_i - \hat{y}_i|$, where N is the number of samples. y_i and $\hat{y}_i$ are the ground truth label and prediction of the i-th sample, respectively. Lower is better.

GM. Error Geometric Mean (GM) is defined as $(\Pi_{i=0}^{N}|y_i - \hat{y}_i|)^{1/N}$, where N is the number of samples. y_i and $\hat{y}_i$ are the ground truth label and prediction of the i-th sample, respectively. Lower is better. We implement GM as $\left(\Pi_{i=0}^{N}\max\{|y_i - \hat{y}_i|, 10^{-10}\}\right)^{1/N}$ for metric robustness.

Spearman correlation. Spearman correlation measures the direction of the monotonic relationship between two variables by calculating the Pearson correlation on their ranked values. We use the implementation in `scipy` library. Higher is better.

B.2 TRAINING DETAILS

For all experiments, models were trained for a maximum of 200 epochs, with early stopping applied if the validation mean absolute error (MAE) did not improve for 30 consecutive epochs. We employed the AdamW optimizer in conjunction with a `ReduceLROnPlateau` learning rate schedule,

which reduced the learning rate by a factor of 0.8 after 5 epochs without improvement. Hyperparameter selection was performed based on validation MAE via grid search, with the trade-off parameter λ of MEX selected from $\{0.25, 0.5, 0.75, 1\}$, batch sizes from $\{32, 64, 128\}$, learning rates from $\{0.00005, 0.0001, 0.001\}$, and weight decay from $\{0, 0.001\}$. All methods were evaluated under three random seeds, and the average and standard deviation of all metrics across all datasets were reported.

B.3 BASELINE DETAILS

In this section, we will introduce the details of all the baseline used in this paper, including LDS, RankSim, BMSE, ConR, C-Mixup, FOMA and ANE.

LDS. LDS (Label Distribution Smoothing) (Yang et al., 2021) convolves a symmetric kernel with the empirical label distribution to account for the continuity of labels. Specifically, we use Gaussian kernel. The algorithm has two hyperparameters—the kernel size and the standard deviation, we set the kernel size as 5 and the standard deviation as 2.

RANKSIM. RankSim (ranking similarity) (Gong et al., 2022) is a regularizer for deep imbalanced regression that enforces an inductive bias where samples closer in label space should also be closer in feature space by aligning the sorted neighbor lists in label and feature spaces. The algorithm has two hyperparameters—the interpolation strength λ and the balancing weight γ. λ trades off the informativeness of the gradient with fidelity to the original function and γ is the balancing weight on the regularization term in the overall network loss. We perform grid search for γ from $\{1, 2, 4\}$ and λ from $\{1, 100, 1000\}$.

BMSE. BalancedMSE (Ren et al., 2022) achieves balanced predictions by utilizing the prior distribution of the training labels to perform a statistical transformation. Specifically, we use BMC (Batch-based Monte-Carlo) to implement this algorithm. BMC does not require prior knowledge of $p_{\text{train}}(y)$. It considers all labels in a training batch as random samples drawn from $p_{\text{train}}(y)$. For a training batch of labels $B_{\boldsymbol{y}} = \{\boldsymbol{y}_{(1)}, \boldsymbol{y}_{(2)}, ...\boldsymbol{y}_{(N)}\}$, the loss is given by:

$$L = -\log \frac{\exp\left(-\left\|\boldsymbol{y}_{\text{pred}} - \boldsymbol{y}\right\|_2^2 / \tau\right)}{\sum_{\boldsymbol{y}' \in B_y} \exp\left(-\left\|\boldsymbol{y}_{\text{pred}} - \boldsymbol{y}'\right\|_2^2 / \tau\right)}, \tag{7}$$

where $\tau = \sigma_{\text{noise}}^2$ is the temperature coefficient and σ_{noise}^2 is a learnable parameter with an initial value of 1.0.

CONR. ConR (Keramati et al., 2024) is a contrastive regularizer designed to capture global and local label similarities in feature space while preventing minority sample features from collapsing into majority neighbors. It identifies mismatches between label and feature spaces, applying penalties to correct them. The algorithm has a temperature hyperparameter τ, we search from $\{0.25, 0.5, 1, 2\}$.

C-MIXUP. C-Mixup (Yao et al., 2022) improves the generalization ability of regression tasks by linearly interpolating a pair of samples and their corresponding labels. Specifically, C-Mixup adjusts the sampling probability based on the similarity of the labels. We use Gaussian kernel to calculate the sampling probability of mixed samples, and the interpolation ratio $\lambda \in [0, 1]$ is drawn from a Beta distribution, i.e., λ ˜Beta(α, α). The algorithm has two hyperparameters—the bandwidth σ of Gaussian kernel and α of Beta distribution, we search for σ from $\{0.1, 1\}$ and α from $\{1, 2\}$.

FOMA. FOMA (First-Order Manifold Augmentation) (Kaufman & Azencot, 2024) is a simple, domain-independent and data-driven DA routine, its core idea is that data with similar dominant components to the training set should be treated as true samples. Let X, Y be the input and output mini-batch tensors, respectively, and $Z_l = g_l(X)$ be the hidden representation at layer l. FOMA generates new training samples $Z_l(\lambda), Y(\lambda)$ from the given ones by scaling down their small singular values with a random $\lambda \in [0, 1]$. λ is drawn from a Beta distribution, i.e., λ ˜Beta(α, α). The algorithm has a hyperparameter— α of Beta distribution, we search it from $\{0.1, 0.5, 1, 10\}$.

ANE. ANE (Automated Nonlinearity Encoder) (Na & Park, 2022) is a data-agnostic embedding technique designed to enhance the extrapolation ability of neural networks. ANE encodes nonlinearities in regression tasks into input embeddings by minimizing the Wasserstein distance between pairwise distances of data samples in both input and target spaces and then optimizes an MLP prediction network with those embeddings.

B.4 MORE EXPERIMENT RESULTS

Table 5: Test GM(↓) on the benchmark dataset.

Model	Algo	Formation Energy	Shear Modulus		Refractive Index		Phonons Mode Peak		Avg Rank
		low	low	high	low	high	low	high	
PaiNN	ERM	0.388(0.001)	0.571(0.094)	0.340(0.001)	0.228(0.008)	0.709(0.020)	0.750(0.006)	0.849(0.020)	7.6
	LDS	0.337(0.019)	0.479(0.001)	0.310(0.003)	0.178(0.003)	0.668(0.008)	0.770(0.026)	0.910(0.098)	5.6
	Ranksim	0.385(0.003)	0.495(0.001)	*0.214(0.030)*	0.223(0.008)	0.659(0)	0.646(0.123)	0.852(0.015)	5.1
	BMSE	**0.247(0.041)**	**0.367(0.041)**	**0.142(0.036)**	0.205(0.074)	*0.501(0.010)*	*0.581(0.018)*	*0.800(0.021)*	2
	ConR	0.367(0.011)	0.489(0.005)	0.306(0.002)	0.275(0.144)	0.66(0.071)	0.688(0.182)	0.814(0.082)	5.4
	C-Mixup	0.35(0.005)	0.491(0.002)	0.325(0.002)	0.212(0.005)	0.67(0.003)	0.752(0.008)	0.852(0.014)	6.3
	FOMA	0.365(0.004)	*0.391(0.067)*	0.284(0.083)	*0.17(0.031)*	0.634(0.019)	0.701(0.008)	0.877(0.012)	3.9
	ANE	0.511(0.006)	0.483(0.002)	0.327(0.003)	0.202(0.006)	0.666(0.022)	0.768(0.03)	0.918(0.026)	6.9
	MEX	*0.278(0.034)*	0.425(0.003)	0.227(0.012)	**0.093(0.01)**	**0.435(0.01)**	**0.555(0.041)**	**0.75(0.022)**	1.7
EquiformerV2	ERM	0.330(0.003)	0.466(0.001)	0.288(0.001)	0.172(0.001)	0.550(0.003)	0.667(0.008)	0.829(0.007)	6.3
	LDS	*0.236(0.01)*	0.446(0.004)	0.275(0.005)	0.136(0.011)	0.555(0.003)	0.686(0.001)	0.806(0.009)	4.1
	Ranksim	0.319(0.002)	0.413(0.089)	0.286(0.003)	0.171(0.002)	0.534(0.001)	0.665(0.008)	0.824(0.003)	4.7
	BMSE	0.301(0.134)	**0.298(0.032)**	**0.11(0.019)**	*0.106(0.001)*	*0.456(0.015)*	*0.389(0.026)*	0.822(0.019)	2.7
	ConR	0.285(0.003)	0.465(0.004)	0.306(0.004)	0.176(0.006)	0.527(0.004)	0.671(0.003)	*0.800(0.01)*	5.4
	C-Mixup	0.277(0.017)	0.463(0)	0.295(0.002)	0.161(0.003)	0.534(0.003)	0.691(0.002)	0.820(0.005)	5.1
	FOMA	0.278(0.005)	0.466(0.001)	0.293(0.001)	0.141(0.006)	0.538(0.002)	0.68(0.007)	0.824(0.001)	5.4
	ANE	0.457(0.019)	0.481(0.001)	0.316(0.002)	0.173(0.008)	0.601(0.013)	0.833(0.035)	0.904(0.013)	8.4
	MEX	**0.122(0.011)**	*0.335(0.017)*	*0.19(0.019)*	**0.056(0.003)**	**0.41(0.03)**	**0.35(0.008)**	**0.668(0.022)**	1.3

Table 6: Test Spearman correlation efficient(↑) on the benchmark dataset.

Model	Algo	Formation Energy	Shear Modulus		Refractive Index		Phonons Mode Peak	
		low	low	high	low	high	low	high
PaiNN	ERM	0.541(0.005)	0.059(0.148)	-0.128(0.047)	-0.116(0.039)	-0.208(0.065)	**-0.181(0.007)**	-0.421(0.004)
	LDS	**0.660(0.013)**	0.171(0.028)	-0.022(0.080)	-0.104(0.024)	-0.265(0.018)	-0.230(0.043)	-0.379(0.018)
	Ranksim	0.542(0.004)	0.170(0.017)	-0.018(0.023)	-0.070(0.0430)	-0.314(0.009)	-0.216(0.046)	-0.418(0.012)
	BMSE	0.351(0.044)	0.099(0.064)	-0.054(0.055)	-0.133(0.066)	-0.237(0.022)	-0.260(0.054)	-0.302(0.017)
	ConR	0.473(0.101)	0.131(0.019)	-0.032(0.033)	0.009(0.029)	-0.174(0.036)	-0.210(0.029)	-0.430(0.018)
	C-Mixup	0.567(0.005)	0.144(0.019)	-0.093(0.017)	-0.052(0.016)	-0.213(0.129)	**-0.181(0.010)**	-0.437(0.014)
	FOMA	0.534(0.013)	0.232(0.020)	-0.046(0.080)	**0.093(0.018)**	-0.325(0.005)	-0.205(0.044)	-0.384(0.042)
	ANE	0.469(0.125)	0.23(0.007)	**0.01(0.027)**	0.084(0.059)	-0.255(0.016)	-0.208(0.059)	**-0.284(0.047)**
	MEX	0.489(0.051)	**0.319(0.015)**	-0.003(0.017)	0.059(0.027)	**0.039(0.039)**	-0.201(0.038)	-0.335(0.018)
EquiformerV2	ERM	0.615(0.015)	0.336(0.042)	0.069(0.024)	0.194(0.015)	-0.120(0.016)	0.008(0.054)	-0.459(0.011)
	LDS	0.71(0.044)	0.155(0.033)	0.020(0.085)	**0.237(0.020)**	-0.031(0.013)	0.009(0.117)	-0.411(0.030)
	Ranksim	0.625(0.014)	0.332(0.009)	0.094(0.016)	0.195(0.021)	-0.060(0.004)	**0.061(0.071)**	-0.432(0.015)
	BMSE	0.273(0.043)	0.278(0.033)	0.074(0.020)	-0.016(0.004)	-0.126(0.0470)	-0.175(0.037)	-0.393(0.003)
	ConR	**0.724(0.012)**	0.357(0.023)	0.109(0.038)	-0.021(0.047)	-0.022(0.008)	-0.050(0.108)	-0.460(0.047)
	C-Mixup	0.682(0.019)	0.328(0.063)	0.117(0.055)	0.183(0.077)	-0.045(0.014)	0.005(0.003)	-0.479(0.008)
	FOMA	0.703(0.004)	0.317(0.032)	0.131(0.060)	0.211(0.028)	-0.012(0.003)	0.040(0.042)	**-0.360(0.184)**
	ANE	0.6(0.053)	0.336(0.006)	**0.171(0.024)**	0.033(0.057)	-0.21(0.053)	-0.232(0.102)	-0.492(0.051)
	MEX	0.645(0.020)	**0.374(0.021)**	0.095(0.027)	0.080(0.027)	**0.088(0.006)**	-0.039(0.085)	-0.427(0.019)