

# Materials Informatics Transformer: A Language Model for Interpretable Materials Properties Prediction

Hongshuo Huang,<sup>†</sup> Rishikesh Magar,<sup>‡</sup> Changwen Xu,<sup>‡</sup> and Amir Barati

Farimani<sup>\*,‡,†</sup>

<sup>†</sup>*Department of Material Science and Engineering, Carnegie Mellon University, Pittsburgh PA, USA 15213*

<sup>‡</sup>*Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh PA, USA 15213*

E-mail: barati@cmu.edu

## Abstract

Recently, the remarkable capabilities of large language models (LLMs) have been illustrated across a variety of research domains such as natural language processing, computer vision, and molecular modeling. We extend this paradigm by utilizing LLMs for material property prediction by introducing our model Materials Informatics Transformer (MatInFormer). Specifically, we introduce a novel approach that involves learning the grammar of crystallography through the tokenization of pertinent space group information. We further illustrate the adaptability of MatInFormer by incorporating task-specific data pertaining to Metal-Organic Frameworks (MOFs). Through attention visualization, we uncover the key features that the model prioritizes during property prediction. The effectiveness of our proposed model is empirically validated across

14 distinct datasets, hereby underscoring its potential for high throughput screening through accurate material property prediction.

## Introduction

Machine Learning (ML) models have made significant progress in the field of computational material science.<sup>1–4</sup> Their applications span from accurate material property prediction<sup>5–15</sup> to novel material generation.<sup>16–21</sup> The integration of ML into materials science provides a possible alternative to traditional computational methods, such as Density Functional Theory (DFT) as they are faster surrogates and can facilitate their adoption as high-throughput screening tools.<sup>22</sup> Notably, Graph Neural Networks (GNNs) stand out as some of the most successful methodologies for material property prediction.<sup>23</sup> Owing to their capability to represent local environments and model atomic interactions, GNNs have considerably advanced the domain of material property prediction.<sup>24</sup> Recent improvements include leveraging angular information to construct the line graph<sup>13</sup> to capture three-body interactions. The coNGN<sup>25</sup> applied nested graphs of atoms, bond angles, and dihedrals. Encoding periodic patterns, Matformer<sup>26</sup> captured the geometric distances between the same atoms in neighboring unit cells. Potnet<sup>27</sup> embedded interatomic potentials, encompassing the Coulomb potential, London dispersion potential, and Pauli repulsion potential.

Despite these considerable strides, GNNs are not without their limitations, GNNs demand relaxed structures as inputs, a requirement that triggers the need for DFT calculations. This requirement escalates computational costs, which grow directly proportional to the size of the crystal, thereby creating a considerable hurdle to scalability for larger systems like Metal-Organic Frameworks(MOFs) or defects properties in supercell crystal.<sup>15</sup> Another challenge faced by GNN models lies in their interpretability. While their message-passing mechanisms are adept at delivering insights at the local atom level by modeling interactions between neighbors. However, they often fail to capture global features especially in capturing

the crystal system, lattice parameter and periodicity.<sup>28</sup> that necessitate an understanding of the overall structural properties in larger crystals. To overcome these challenges, various computational strategies have been explored such as the coordinate free approach in Wren,<sup>29</sup> or multi model training. One other approach involves the use of structure-agnostic methods, which aim to bypass the need for specific structural data. These models use formulas as inputs, which helps lower computational requirements.<sup>29–31</sup> However, they tend to be less accurate and have difficulty distinguishing between the geometries of different crystalline structures.

One other novel approach is to represent structures as text and use Large Language Models (LLMs), which has emerged as a promising solution in artificial intelligence. LLMs have been successful in a variety of natural language processing tasks and have shown potentials in molecular machine learning tasks.<sup>10,32–38</sup> The success in predicting properties of proteins and molecules can be credited to the use of straightforward representations like amino acid sequences (FASTA)<sup>39</sup> representing either nucleotide sequences or amino acid sequences and SMILES<sup>40</sup> describing the structure of chemical species. Both FASTA and SMILES are text-based representations, making them naturally compatible with LLMs designed to infer and generate text and can be easily integrated into the transformer architecture. Furthermore, LLMs offer a degree of interpretability as the attention heads can be visualized, providing insights into the most informative tokens that the model deems critical for downstream tasks. The success of language model in molecular area indicate a potential avenue for representing crystalline structures in a manner suitable for transformer models. The ability to convert intricate three-dimensional structural information of materials into tokenized sequences could facilitate the application of LLMs in material property prediction. However, there's no comparable comprehensive representation for crystalline materials that can be easily integrated into the transformer architecture. To leverage the benefits of LLMs in this area, it is imperative to create a representation of crystalline materials. Once this representation is developed, the subsequent challenge is of developing tailored pretraining methods

in the context of crystalline materials.

Devising appropriate pretraining methods is fundamental when developing LLMs to predict material properties. Pretraining plays a pivotal role in bolstering the effectiveness of deep learning models. In the context of computational material science, this stage is even more critical due to the sheer complexity and diversity of the materials landscape.<sup>41</sup> While GNN models have seen approaches such as Node and Edge Prediction, Self-supervised Learning,<sup>9,10</sup> and Graph Contrastive Learning,<sup>42,43</sup> but these method is not suitable for LLMs which typically using Masked Language Modelding (MLM).

This leads us to the following questions: 1.) Can we leverage the inherent crystallographic information such as space group as input representation for the LLMs? 2.) Given an appropriate representation, can we develop a transformer architecture that can accurately predict material properties whilst offering a degree of interpretability of input tokens and flexibility to apply? 3.) Can we develop proper pretraining methods for such transformer architecture to learn crystallography and capture a broad landscape of materials space ?

In this work, we introduce Materials Informatics Transformer (MatInFormer) that attempts to address some of the aforementioned questions (Figure 1). We formulate a robust textual representation for crystalline materials, The architecture is capable of learning about the crystalline system by tokenizing the space group information that can in some ways capture the geometry of the crystals, informatics tokens depending on type of crystalline material, and formula tokens that capture the composition of the crystalline material(Figure 1a). Using the information from the space group, informatics, and formula tokens we feed it into the MatInFormer model based on the Roberta architecture.<sup>44</sup> We pretrain the MatInFormer using three different strategies. For the first strategy, we pretrain the MatInFormer(Figure 1b) using the classic Masked Language Modeling(MLM). This enables the model to capture the unique characteristics of the crystal system. Additionally, we also pretrain the model predicting the lattice parameters. Finally, we combine the MLM and lattice parameter prediction and use it for pretraining the model. The pretrained model is later

finetuned for different downstream tasks(Figure 1c). Using the MatInFormer architecture we predict the properties of the material for a diverse range of materials from the Matbench suite.<sup>45</sup> Furthermore, we also investigate the efficacy of the model for property prediction of Metal-Organic Frameworks(MOFs).<sup>46,47</sup> We demonstrate the interpretability aspect of the models by analyzing how different tokens influence the performance of the model. Moreover, we also demonstrate the flexibility of our framework by easily manipulating the number of tokens that are input into the transformer model and illustrating how they influence the performance of the model. Our model MatInFormer bridges LLMs with computational material science, presenting an interpretable framework for property prediction.

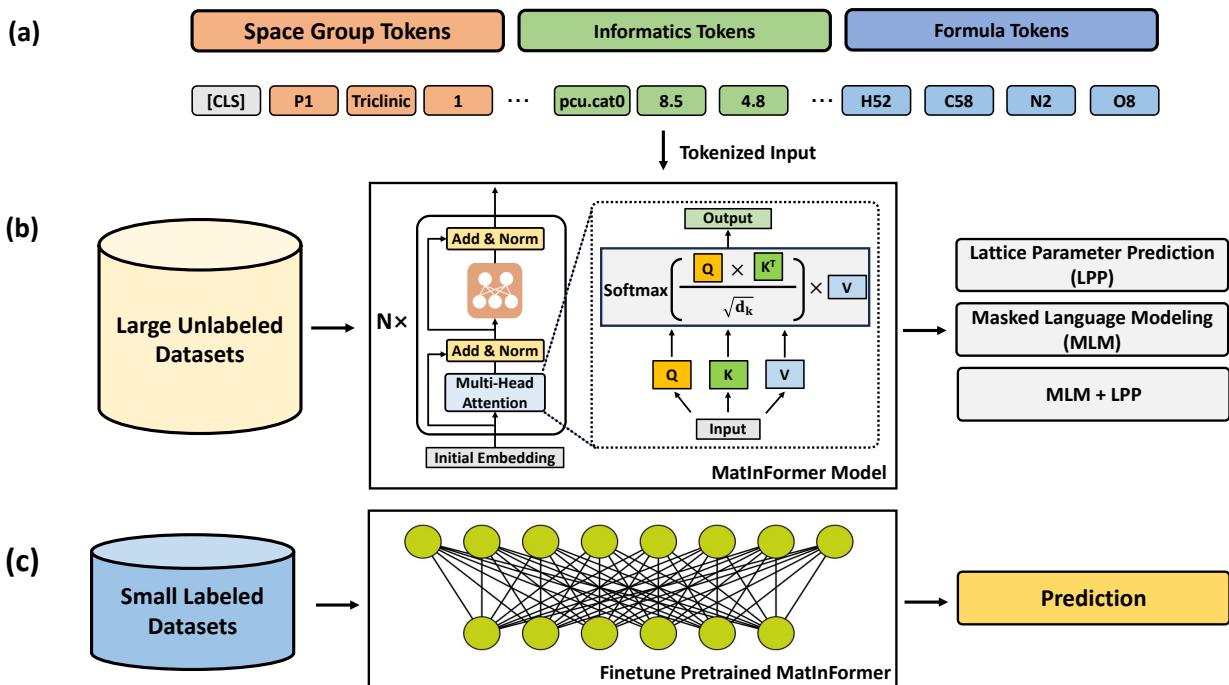


Figure 1: Framework of Material Informatics Transformer. **(a)**The tokens used for representing material, including Space Group Tokens, Informatics Tokens, and Formula Tokens. The representations are tokenized and combined as the input of the Transformer model. **(b)** The pretrainning stage. A Large unlabeled dataset is used to pretrain the transformer encoder with 3 different pretrain method. **(c)** Fintuning stage for the downstream tasks. The pretrained transformer is then finetuned for different labeled material property prediction tasks.

# Methods

## Materials Informatics Tokenization

The initial input to the MatInFormer model is an embedded representation produced through the tokenization of available material information. This tokenization process for the MatInFormer model divides the material representation into three distinct parts: space group tokens, informatics tokens, and formula tokens.

The first of these is the space group tokens. In the field of crystallography, a crystalline material structure is specified by a combination of the structure's space group, the dimension of its unit cell, and the element at its Wyckoff position.<sup>48,49</sup> To employ a coordinate-free approach, we discard precise atomic positions, lattice parameters, and Wyckoff position, retaining only the space group as the input. There are 230 space groups in total, which combine 32 crystallography point groups with 14 Bravais lattices. We have standardized the sequence length for space group tokens to 12, as shown in Table 1. The space group is mapped using a dictionary to ensure a comprehensive representation of group-specific information. This includes the space group symbol, point group, crystal system, and their corresponding group properties. For instance, the space group tokens for *Fm* $\bar{3}m$  (No. 225) are  $[F\frac{4}{m}\bar{3}\frac{2}{m}]$ , [225], [192], [ $m\bar{3}m$ ], [cubic], [ $m\bar{3}m$ ], [Centrosymmetric], [non-polar], [F], [ $\frac{4}{m}$ ], [ $\bar{3}$ ], [ $\frac{2}{m}$ ]. The space group token representation follows the exact order as specified in Table 1. We calculated the space group symbol using Pymatgen's SpacegroupAnalyzer. This analyzer takes threshold distance and angle tolerance as parameters to determine the Space Group. To determine the space group, we used the default settings with a threshold distance of 0.01 $\text{\AA}$  and an angle tolerance of 5.0°, respectively. We would like to note, that these default settings in Pymatgen are slightly different from those in the Materials Project,<sup>22</sup> where the tolerance for distance is set at 0.1 $\text{\AA}$ . In real material systems, there's often a degree of off-site relaxation from the high-symmetry sites, leading to some materials being classified as "P1". However, this distance tolerance only marginally impacts the model's performance

on downstream tasks, which aligns with findings from previous studies, such as Wren.<sup>29</sup>

Table 1: Space group and Informatics Tokens for MatInFormer

Token Index	Token
0	Full Space Group Symbol
1	Space Group Number
2	Order
3	Point Group
4	Crystal System
5	Laue Class
6	Symmetry
7	Polar
8	Separate Space Group Token 0
9	Separate Space Group Token 1
10	Separate Space Group Token 2
11	Separate Space Group Token 3
12	Topology
13	Unit Cell Volume
14	Atoms Number
15	Porosity Fraction
16	Accessible Void Fraction

The second component of the tokenized input involves informatics tokens. These capture essential attributes associated with a particular type of crystalline system. Consider, for instance, Metal-Organic Frameworks (MOFs). To distinguish between different structures within the hypothetical MOF (hMOF) dataset, relying solely on the formula and space group proves insufficient. More than 53% of the samples in the hMOF database have duplicate formula and space group. If we use only the space group and formula tokens, the model will determine these to be the same materials. However, those materials have different 3D structure which leads to them having a different labels. As such, we turn to other tokens that can provide more information that distinguishes the MOF in question. The first such token is topology, drawn from the MOFid as per Bucior et al. (2019).<sup>50</sup> Every MOFid includes topology and catenation codes adopted from the Reticular Chemistry Structure Resource (RCSR) database.<sup>51</sup> In addition to topology, we tokenize the unit cell volume and the number of atoms. This allows the model to potentially learn porosity features through the

formula and volume. To guarantee the model’s capacity to capture these porosity features, we tokenize the porosity fraction and the accessible void fraction as calculated by porE.<sup>52</sup> To calculate the porosity we apply the Grid Point Approach (GPA). According to GPA, a numerical grid near any atom (within the van der Waals sphere) is considered occupied, while any other grid is seen as unoccupied. Consequently, the void fraction is calculated using Equation S1:

$$\Phi_{void} = \frac{N_{unoccupied}}{N_{total}} \cdot 100\% \quad (1)$$

where  $\Phi_{void}$  represents the void fraction,  $N_{unoccupied}$  represents the number of unoccupied grids, and  $N_{total}$  signifies the total number of grids. The accessible porosity is estimated under the premise that a sphere with a probe radius ( $r_{probe}$ ) does not interact with the atoms. Additional details about the GPA approach that we used for porosity calculation are available in SI(Figure S5).

Taking hybrid organic-inorganic perovskites (HOIPs) as another example, we treat the organic cation as an informatics token. The application of informatics tokens lends a level of flexibility to the architecture. This flexibility can be used to incorporate additional information and simultaneously minimize redundancy.

The third component of our tokenization process is the formula tokens. The stoichiometric formula is used to generate an elemental information embedding using "Matscholar" embedding.<sup>53</sup> The elemental information has dimensionality of  $d_{el} = 200$ , concatenated with each element in the composition weighted by its respective elemental fraction. Finally, the formula embeddings shape in  $(N, 201)$  where N represents the number of the formula tokens(also the number of the elements). We standardize the sequence length of the formula tokens to 20, which corresponds to the maximum number of elements observed in our dataset. If a particular stoichiometric formula does not contain 20 elements, we employ zero-padding to achieve a consistent length across all formula tokens.

Finally, to ensure compatibility across all token types, the formula tokens are projected

into the appropriate dimension using a single linear layer. Following this transformation, all three types of tokens—formula, space group, and informatics—are simultaneously concatenated to form a unified token set. This aggregated set is then fed into the transformer, offering a comprehensive representation of the crystalline material system.

## Transformer encoder

Our MatInFormer is based on a Roberta transformer architecture<sup>44</sup> from Huggingface<sup>54</sup> library. The MatInFormer model employs self-attention layers as its primary method of operation using "Scaled Dot-Product Attention". The input data are mapped into 3 different embeddings to calculate the attention: queries(Q), keys(K), and values(V). And the attention score is calculated

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2)$$

where  $d_k$  is the dimensions of  $K$ .

Materials Informatics Transformer is able to capture bidirectional information meaning it can processes the input data from both left-to-right and right-to-left directions and is able to capture features from both directions. The transformer encoder with 8 blocks, 12 heads, and 768 hidden sizes is used. A [CLS] token is located at the first position in each layer. The [CLS] embeddings of the last layer are used for the downstream task.

## Pretraining

Pretraining significantly enhances the learning capabilities of LLMs, especially when trained on vast unlabeled datasets. Our pretraining datasets are sourced from JARVIS-Tools,<sup>55</sup> which consists of six distinct datasets.<sup>56</sup> The first dataset, curated from five out of the six datasets from JARVIS-Tools (excluding the Open Catalyst dataset<sup>57</sup>), contains 642,459 data samples. The second dataset combines all six datasets, including the Open Catalyst dataset,

totaling 997,528 samples. The dataset with pretraining size 642,459 is named as pretraining dataset 1 and dataset of size 997,528 is named as pretraining dataset 2. Additional details about the pretraining datasets are available in Table S1. Moreover, we also explore crystal system distribution(Figure S1 and Figure S3) and elemental distribution(Figure S2 and Figure S4) for both the pretraining datasets. In this work, we have developed three distinct pretraining strategies namely masked language modeling(MLM), Lattice Parameter Prediction(LPP), and a combination of MLM and LPP.

### **Masked Language Modeling (MLM)**

Our first strategy leverages the Masked Language Modeling (MLM) objective for pretraining the MatInformer. We employ a masking ratio of 25% for space group tokens during pre-training, thereby enabling the model to learn and infer the masked space group information. This process enables the Materials Informatics Transformer to understand the intricate crystallographic relationships among 'Space Group', 'Point Group', and 'Crystal System' tokens. Under the MLM strategy, we task the model with predicting the masked space group tokens, thereby enabling it to learn crystallographic information. This includes the classification of the crystal system based on a given space group, or the prediction of the space group number from individual space group tokens. The model is pretrained using the cross-entropy loss function and trained for 50 s under the MLM objective. The pretraining hyperparameters for the MLM strategy are given in Table S3.

### **Lattice Parameter Prediction (LPP)**

The second strategy involves lattice parameters prediction(LPP) of the unit cell (namely,  $a, b, c$  and  $\alpha, \beta, \gamma$ ). This approach empowers the model to understand the crystallography interplay among space groups, formulas, and lattices. Unlike the first strategy of MLM, this technique obliges the model to not only understand crystallography and group symmetry, but also to incorporate formula and elements in order to reconstruct the coordinates. In

contrast to masked language modeling, this approach enables MatInFormer to learn via crystal system characterization, relationships between lattice parameters, and also take into account the composition of the material. For example, if the crystal system token indicates a [hexagonal] system, the model needs to predict the lattice parameters adhering to the geometric characteristics of a hexagonal system ( $a = b \neq c$ ,  $\alpha = \beta = 90^\circ$ ,  $\gamma = 120^\circ$ ). For the pretraining process, we utilized multi-task prediction, implementing a Multilayer Perceptron (MLP) regression head with six outputs for all the lattice parameters. The pretraining hyperparameters for the LPP strategy are given in Table S3.

### **Masked Language Modeling and Lattice Parameter Prediction**

For the third pretraining strategy, the MLM and LPP methods combined together. We mask 25% of the space group tokens and the model is required to predict the lattice parameter using rest of the tokens. Using such a strategy we are able to capture the relationships between space group tokens in addition to using the available crystal information that has not been masked to predict the lattice parameters. The pretraining hyperparameters for the MLM + LPP strategy are given in Table S3.

### **Training Details**

In the pretraining stage, the transformer encoder is coupled with a 2-layer MLP regression head. Depending on the pretraining objective we choose the loss function. For the MLM objective, we use the cross entropy loss function. For lattice parameter prediction, the model is trained to minimize the Mean Square Error(MSE) between prediction and the target lattice parameters. Additional details about the hyperparameters used for pretraining the model are available in SI.

## Finetuning

To evaluate the performance of MatInFormer and ascertain the efficacy of our pretraining strategies, we fine-tuned it on several downstream property prediction tasks. We assessed MatInFormer’s performance using the Matbench suite, as well as the HOIP,<sup>58</sup> hMOF<sup>47</sup> datasets. During fine-tuning, our primary inputs were the ”Formula” and ”Space Group” tokens. Depending on the specific dataset, we supplemented these with informatics tokens.

When fine-tuning, we utilized the weights from the pretrained transformer encoder. For a comparative analysis, we also benchmarked against a model trained from scratch, that was initialized with random weights. For the final property prediction, we used the [CLS] token coupled with a prediction head consisting of 2 MLP-layers with SiLU<sup>59</sup> activation.

For the Matbench tasks, we adhered to the standard 5-fold cross-validation protocol. For the hMOF dataset, we maintained a 70%/15%/15% train/validation/test split, mirroring the approach taken by MOFormer. Here, the additional informatics tokens included topology, volume of the unit cell, number of atoms, and porosity. Furthermore, we benchmarked against the HOIP dataset,<sup>58</sup> employing an 80%/20% split for training and testing, respectively. In the case of the HOIP dataset, we introduced the organic group as an extra informatics token. Detailed finetuning hyperparameters<sup>60,61</sup> can be found in the supplementary information (SI)(Table S4).

## Results and Discussion

To evaluate the model performance on materials property prediction, we use 8 datasets from the Matbench suite. Moreover, we also benchmark hMOF datasets about  $CO_2$  and  $CH_4$  absorption for different pressure settings, shown in Table 2. The selected datasets not only cover an extensive spectrum of properties such as exfoliation energy, phdos peak, refractive index, modulus, formation energy, band gap, and gas absorption but also vary from each other a lot in terms of the size of the datasets, ranging from 636 to 132,752. We compare

our the performance of our model on Matbench to other structure agnostic models: Roost and Finder; Structured-based GNN models: ALIGNN and CGCNN; Coordinate-free model: Wrenformer. Among the baselines, Wrenformer is primarily used for comparison since we use the same transformer architecture and share the same level of coordinate-free information. And for hMOF, we compare our result to CGCNN, MOFormer, and Stoichiometric-120. and hMOF result is shown in Table 4. The results for the standalone HOIP dataset are shown in Table S5.

Table 2: Overview of the datasets used for benchmarking the performance of the Pretraining framework. We predict the properties on 8 different datasets<sup>22,58,62–68</sup> aggregated from the Matbench suite<sup>45</sup> and hMOF.

Dataset	# Samples	Property	Unit
JDFT2D (JDFT) <sup>62</sup>	636	Exfoliation Energy	meV per atom
Phonons <sup>63</sup>	1,265	Last Phdos Peak	1 per cm
Dielectric <sup>64</sup>	4,764	Refractive Index	Unitless
GVRH <sup>65,66</sup>	10,987	Shear Modulus	$\log_{10} GPa$
KVRH <sup>66</sup>	10,987	Bulk Modulus	$\log_{10} GPa$
Perovskites <sup>67</sup>	18,928	Formation Energy	eV per atom
MP-Gap (MP-BG) <sup>22</sup>	106,113	Band Gap	eV
MP-E-Form (MP-FE) <sup>22</sup>	132,752	Formation Energy	eV per atom
hMOF <sup>47</sup>	102,858	Gas Absorbtion	$mol * kg^{-1}$
HOIP <sup>58</sup>	1,333	Band Gap	eV

## Matbench

The results for Matbench are shown in Table 3. We compared the performance of MatInFormer, trained from scratch, with that of Wrenformer. Wrenformer uses a coordinate-free transformer based on the Wyckoff representation. Out of 8 tasks, MatInFormer performed better than Wrenformer in 6. Notably, MatInFormer predicted the phonons dataset with 47.0% lower MAE than Wrenformer did. Also, MatInFormer showed an average reduction of 11.87% in MAE across the smaller datasets of jdft2d, phonons, dielectric, and modules. There was a 6.93% decrease in MAE for the band gap and formation energy datasets. Although

MatInFormer has an architecture similar to Wrenformer, it still performed better without using detailed Wyckoff positions and representations. This improvement suggests that our model benefits from the Roberta encoder. Additionally, tokenizing space group tokens to provide crystallography information and symmetry properties directly enhances the model’s performance. These results indicate that MatInFormer has some state-of-the-art results for coordinate-free models. When we compare MatInFormer to structure-agnostic models like Roost, the performance improvement is clear. MatInFormer showed improvements of 12.62% for GVRH, 11.71% for KVRH, and 23.63% for formation energy.

We also pretrained the MatInFormer using three different strategies. We observed that when MatInFormer was pretrained using Masked Language Modeling (MLM), its performance declined compared to when trained from scratch for some datasets. During the MLM pretraining, random space group tokens are masked and the model is trained to predict those mask tokens with other space group tokens. For example, the models need to predict the crystal system ”Orthorhombic” and point group ”mm2” for the given space group symbol ”Pmm2”. However, formula tokens will stay the same since there is no way to predict the formula for a given space group and result in only crystallography information being captured by the model. We noticed that the MLM objective did not help improve performance greatly on downstream tasks. One possible reason for this might be that the space group tokens mainly focus on the geometric properties of the crystal system, and don’t really capture the crystalline materials composition. Given the observed limitations with MLM, we then pretrained the model via Lattice Parameter Prediction (LPP), where the model has to not only learn the information about the rules of the crystal system but also needs to consider the impact of formula tokens. Finally, aiming to leverage the strengths of both approaches, we adopted a hybrid strategy that combined both MLM and LPP for pretraining.

We found that employing the LPP and MLM + LPP pretraining strategies significantly enhanced the performance of the baseline MatInFormer. For instance, using the LPP strategy led to an average improvement of 5.36%, with the jdft2d, phonons, and band gap datasets

showing notable improvements of 13.28%, 11.83%, and 8.09%, respectively. Similarly, the MLM + LPP strategy yielded an average improvement of 4.45%, with the maximum performance gains again observed in the jdft2d, phonons, and band gap datasets. However, when we pretrained another model using the LPP strategy on a larger dataset with OCP<sup>57</sup> data, the average improvements reduced to 2.36%. This decrease could potentially be attributed to the presence of unrelaxed structures in the OCP data, which implies that the lattice parameter values are suboptimal, making pretraining to predict them less beneficial.

When comparing the pretrained MatInFormer to structure-based models like CGCNN, MatInFormer outperforms in 6 out of 8 tasks after being pretrained using Lattice Parameter Prediction. Unlike CGCNN, MatInFormer processes only the space group tokens and the formula tokens. This indicates that there is potentially a lot of information for models to tap into, beyond just atom distances, angular data, and the extensive feature engineering employed by many structure-based GNNs. The space group captures the patterns and symmetries in crystallography, which some GNN models may not effectively harness.

Even though MatInFormer achieves performance comparable to GNNs for some tasks, the performance gap between structure-based GNNs and other models lacking atomic distance and angle information is stark, particularly for Perovskites dataset.. The MAE of our model aligns with that of structure-agnostic models like Roost. MatInFormer’s weaker performance suggests that perovskite formation energy might depend more on detailed structural attributes, such as atomic positions, than on just formulas or elements. While Wrenformer slightly outperforms MatInFormer by incorporating additional Wyckoff position embeddings, the MAE for these models is still notably higher than that of structure-based GNNs

## Effect of pretraining dataset size

The size of the pretraining data directly influences the performance on downstream tasks. We evaluated the downstream performance using two distinct pretraining datasets. The first dataset contains 642,459 samples, while the second, larger one comprises 997,528 samples.

Table 3: MAE of structure-based GNN, coordinate-free transformer and structure agnostic GNN of Matbench materials property prediction. The best performing result among the coordinate free and structure agnostic approaches has been shown in boldface and next best performing result has been underlined. We have rounded off the results to the third significant digit after the decimal wherever necessary.

Models	jdft2d	phonons	dielectric	GVRH	KVRH	Perovskites	Band Gap	E form	Type
ALIGNN	43.42±8.94	29.53±2.11	0.3449±0.0871	0.072±0.001	0.057±0.003	0.029±0.001	0.186±0.003	0.022±0.001	Structure-based
CGCNN	49.24±11.58	57.76±12.31	0.599±0.083	0.090±0.002	0.071±0.003	0.045±0.001	0.297±0.004	0.034±0.001	Structure-based
MatInFormer	49.61±14.02	48.85±8.71	0.327±0.083	0.090±0.001	0.070±0.003	0.407±0.008	0.290±0.005	0.065±0.001	Coordinate free
MatInFormer - MLM	47.94±13.50	56.72±7.28	0.334±0.074	0.091±0.001	0.072±0.002	0.418±0.006	0.280±0.003	0.066±0.066	Coordinate free
MatInFormer - LPP	<u>43.01±11.83</u>	<u>43.07±4.31</u>	<u>0.322±0.085</u>	<u>0.087±0.002</u>	<b>0.069±0.003</b>	0.409±0.007	0.267±0.002	<b>0.064±0.000</b>	Coordinate free
MatInFormer - LPP(OCP)	43.07±12.99	43.18±5.27	0.338±0.071	<b>0.086±0.001</b>	<u>0.070±0.004</u>	0.409±0.007	0.300±0.008	0.067±0.001	Coordinate free
MatInFormer MLM + LPP	44.02±11.05	<b>42.57±5.18</b>	<b>0.317±0.088</b>	0.087±0.002	0.070±0.004	0.427±0.010	0.265±0.002	<u>0.065±0.001</u>	Coordinate free
Wrenformer	<b>39.63</b>	92.33	0.3583	0.1070	0.0811	<b>0.3350</b>	0.2986	0.0694	Coordinate free
Roost	44.64±11.73	54.39±4.73	0.325±0.078	0.103±0.002	0.080±0.004	<u>0.403±0.008</u>	<u>0.257±0.006</u>	0.085±0.002	Structure agnostic
Finder	47.96±11.67	46.57±3.74	0.320±0.081	0.100±0.002	0.076±0.003	0.645±0.017	<u>0.231±0.003</u>	0.084±0.001	Structure agnostic

The added difference in the second dataset is the inclusion of OCP<sup>57</sup> data. To understand the impact of dataset size on performance, we pretrained two models using the LPP strategy and assessed their downstream performance. We observed that models pretrained without the OCP data showed better results. As detailed in Table 3, the LPP model that did not utilize the OCP dataset consistently outperformed the other model. One of the possible reason for this might be that the Open Catalyst data contains unrelaxed structures. These structures, have lattice parameters that are not in line with the relaxed structure crystallography rules and could introduce complexities in feature capture.

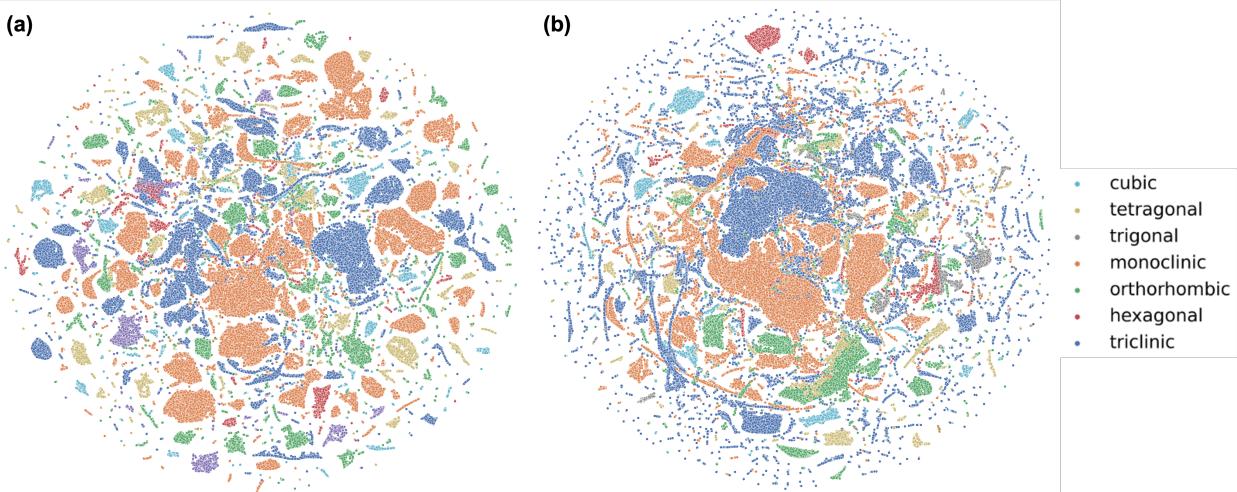


Figure 2: t-SNE plots for two different pretrain datasets

## Exploring the representation space

To explore the representations learned by the pretrained models, we visualized the representation learned by the pretrained models using lattice parameter prediction strategy. To plot these representations in a 2-D space, we applied t-SNE<sup>69</sup> dimension reduction on the CLS token from the last encoder layer. The t-SNE plots are shown in Figure 2.

To better understand the patterns in the t-SNE plots, we note that materials primarily cluster based on their crystal system. This suggests that our model can roughly classify crystal structures using just the formula and space group. However, several scattered points appear around the edges of the t-SNE plots. This scattering can be attributed to the inherent difficulty of predicting lattice parameters for models that don't rely on coordinates. The challenge is amplified given that our pretraining datasets range from simple substances, like pure metals, to complex structures like hMOFs, which can contain between 1 to 20 elements and have unit cell volumes that can be hundreds to thousands of times larger.

Comparing two different plots for the pretraining dataset 1 and pretraining dataset 2, we observed that the datasets pre-trained with Open Catalyst Data exhibited more scatter points on the periphery, whereas most of the data clustered in another plot. This could potentially be attributed to the large amount of data from Open Catalyst having the same formula and space group but varying structures, as some of them are distorted structures. This phenomenon is not only evident in the t-SNE plots but also results in inferior performance on the downstream task, as indicated in Table 3. This explanation, however, is one of several possibilities, and the increased scattering observed may also be influenced by other factors.

## Metal Organic Frameworks(MOFs)

We assessed the performance of MatInFormer for gas absorption in hMOF using various informatics tokens and compared its results to those of CGCNN, MOFormer, and Stoichiometric-120. To understand the influence of different informatics tokens on the model's efficiency, we

conducted ablation studies specifically for hMOF. The results (MAE) for gas adsorption of hMOF are presented in Table 4. MatInFormer, when used without any additional informatics tokens, only performs marginally better than Stoichiometric-120. With the introduction of topology as an informatics token, MatInFormer(T) sees a 24.42% enhancement in performance, aligning it closely with MOFormer which uses mofid as input. When we incorporate informatics tokens related to unit cell volume and the number of atoms, MatInFormer(TV) boosts its performance by 51.86%, surpassing CGCNN. Further, when porosity is also considered, MatInFormer(TVP) achieves an improvement of 53.44% and stands out as the top performer for hMOF gas absorption. We note that the performance of MatInFormer(TVP) is similar to that of MatInFormer(TV).

To highlight how MatInFormer compares with other baseline models, we plotted the MAEs of different models based on their average performance in Figure 3. We observe that the performance ranking of the models follows the amount of information they use for making the predictions. In the first tier, both MatInFormer and Stoichiometric-120 mainly use formulas. Since over 95.8% of hMOF’s space group is P1, the formula becomes the main difference. But because most hMOFs have common elements like C, N, O, H, and a few metals, it’s hard for these models to distinguish between different MOFs effectively. In the second tier, we have the MatInFormer(T) and which adds topology information helps its performance. Meanwhile, MOFormer does even better because it uses bond information from SMILES in its mofid representation. It’s important to note that a lot of the topologies, more than half, are labeled ”pcu.cat0” or ”pcu.cat1”. Even with the help of topology, the model struggles to differentiate more than half of the MOFs. In the third tier, CGCNN uses detailed atomic positions and coordinates, making it really good at telling MOFs apart. But interestingly, MatInFormer versions, specifically MatInFormer(TV) and MatInFormer(TVP), still do better than CGCNN, even though they use less detailed information such as exact atomic coordinates. This shows that MatInFormer is really good at figuring out features like porosity just from cell volume and formula.

Table 4: MAE values of hMOF absorption. MatInFormer(T) denotes topology as an informatics token, MatInFormer(V) denotes unit cell volume and number of atoms as an informatics token, MatInFormer(P) denotes porosity as an informatics token.

Models	$CO_2$ 0.05bar	$CO_2$ 0.5bar	$CO_2$ 2.5bar	$CH_4$ 0.05bar	$CH_4$ 0.5bar	$CH_4$ 2.5bar
CGCNN	<b><math>0.126 \pm 0.005</math></b>	<b><math>0.391 \pm 0.017</math></b>	$0.818 \pm 0.050$	$0.028 \pm 0.001$	$0.121 \pm 0.006$	$0.333 \pm 0.017$
MOFormer	$0.178 \pm 0.002$	$0.558 \pm 0.001$	$1.000 \pm 0.013$	$0.034 \pm 0.000$	$0.174 \pm 0.002$	$0.385 \pm 0.003$
Stoichiometric-120	$0.282 \pm 0.002$	$0.983 \pm 0.005$	$1.895 \pm 0.003$	$0.05 \pm 0.001$	$0.269 \pm 0.001$	$0.631 \pm 0.002$
MatInFormer	$0.245 \pm 0.003$	$0.933 \pm 0.006$	$1.848 \pm 0.005$	$0.038 \pm 0.001$	$0.238 \pm 0.004$	$0.599 \pm 0.006$
MatInFormer(T)	$0.194 \pm 0.001$	$0.661 \pm 0.007$	$1.261 \pm 0.007$	$0.032 \pm 0.001$	$0.186 \pm 0.002$	$0.443 \pm 0.005$
MatInFormer(TV)	$0.129 \pm 0.001$	$0.403 \pm 0.009$	$0.691 \pm 0.007$	$0.022 \pm 0.000$	$0.120 \pm 0.003$	$0.290 \pm 0.005$
MatInFormer(TVP)	$0.131 \pm 0.002$	$0.398 \pm 0.007$	$0.672 \pm 0.007$	$0.021 \pm 0.000$	$0.113 \pm 0.002$	$0.270 \pm 0.005$

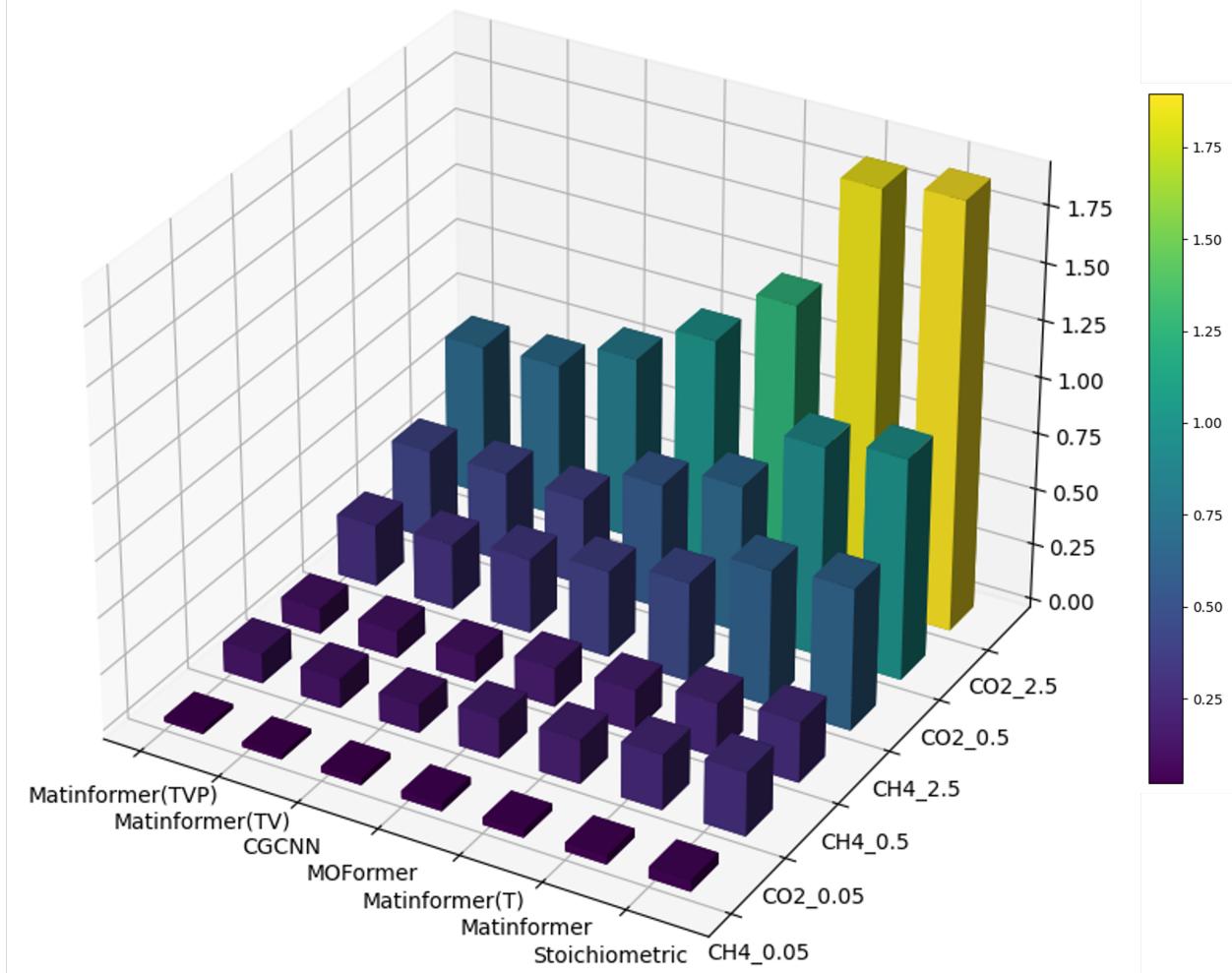


Figure 3: MAE for different models on MOF property prediction tasks. We compare the MatInFormer with models like CGCNN, MOFormer, and Stoichiometric -120. The result for baselines are obtain from MOFormer paper.<sup>10</sup>

## Visualizing the Attention

To gain further insights into our model, we analyzed the attention patterns by visualizing attention scores for a randomly chosen example, hMOF-5035957. These scores, shown as

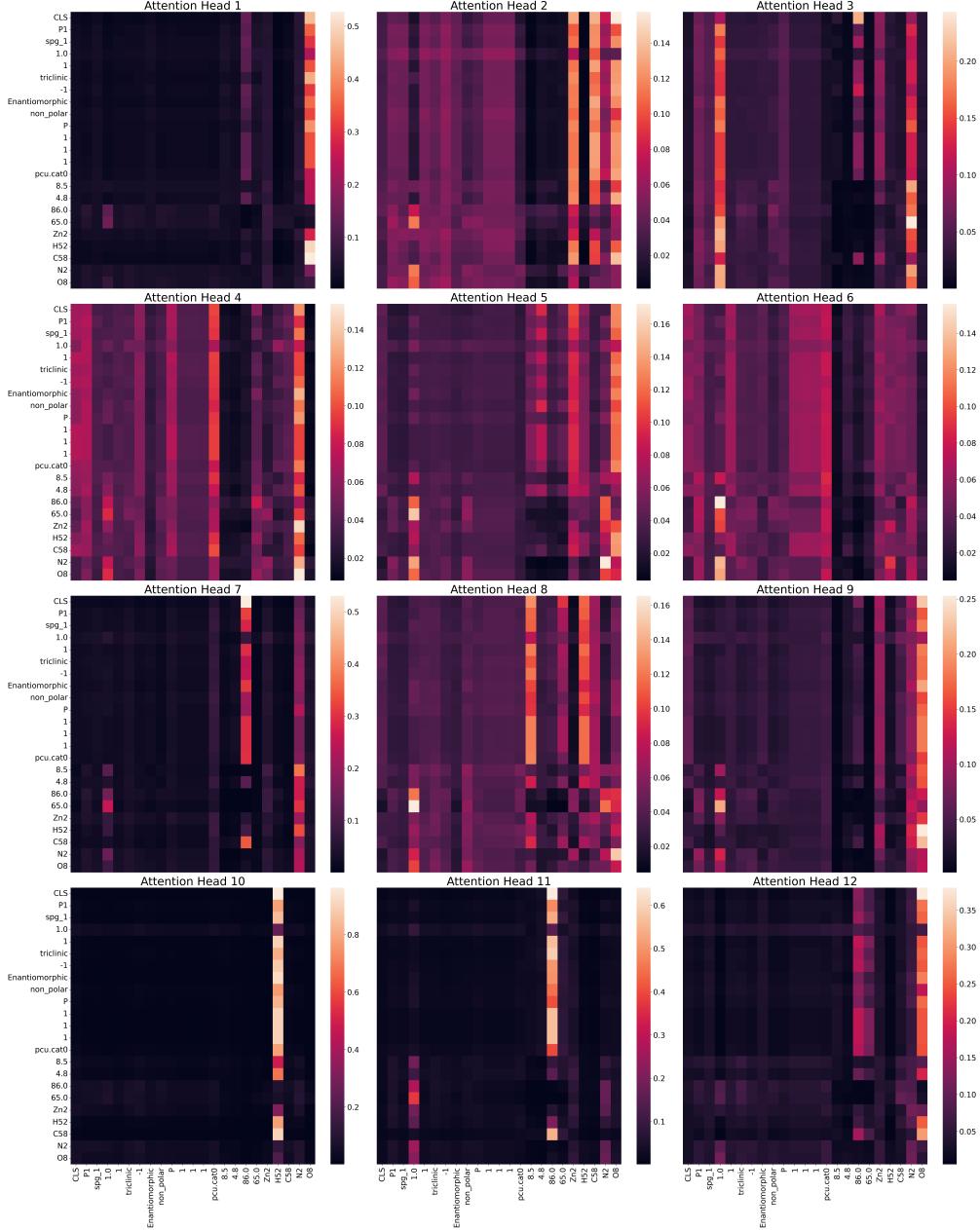


Figure 4: The attention weights for the last encoder layer of the model. Visualizing the attention weights gives a degree of interpretability to the MatInFormer model.

a heatmap in 4, indicate the correlations between tokens for the last layer of the encoder. We observe that some tokens have particularly strong attention signals. Heads 1, 10, and 12, for instance, heavily focus on the formula tokens. Moreover, heads 7 and 11 seem to focus to porosity fractions. Additionally, some of the heads distribute their attention across the space group tokens or evenly spread their attention across all tokens, capturing a more

generalized representation of the material. For the downstream tasks, the regression head is coupled with output of the [CLS] tokens. We further analyzed the attention weights of the [CLS] token, the detailed visualization is shown in Figure S6.

## Conclusion

In this study, we developed MatInformer, a coordinate-free transformer framework designed for materials property prediction. MatInformer achieves excellent performance on many different property prediction tasks even without explicitly considering structural information. Another advantage of our model lies in its ability to maintain consistent memory use, regardless of the growth in crystal size, similar to other structure-agnostic models. The pre-training strategies that we devised such as predicting lattice parameters, enables the model to gain insights into crystallography, leading to marked improvements in downstream tasks. When applied to hMOF, we investigate the influence of different features through an ablation study, considering informatics tokens like topology, unit cell volume, and porosity. The use of informatics tokens for MOFs further demonstrates the flexibility and potential of the MatInformer to incorporate material information through simple tokenization. Moreover, the visualization of attention heatmaps provides a degree of interpretability for the model. This factor of interpretability might offer insights into how deep learning models generate material representations.

Although the MatInformer shows promising results there are multiple challenges ahead. The Transformer model inherently will have difficulties in tokenizing numerical values such as atomic distances and angles, which are crucial for structure-based regression tasks. There is a potential to develop materials specific tokenization strategies. Additionally, multimodal approaches could possibly be leveraged to maximize the benefits inherent to both structure based and structure agnostic models. Overall, we believe MatInformer is a promising step toward harnessing the power of LLMs to facilitate material discovery.

## Data availability

All Matbench data are available on Matbench Website and Pretrain data through JARVIS Databases

## Code availability

The python code of this work can be found on Github <https://github.com/hongshuhu/MatInFormer>

## References

- (1) Schmidt, J.; Marques, M. R.; Botti, S.; Marques, M. A. Recent advances and applications of machine learning in solid-state materials science. *npj Computational Materials* **2019**, *5*, 83.
- (2) Pilania, G. Machine learning in materials science: From explainable predictions to autonomous design. *Computational Materials Science* **2021**, *193*, 110360.
- (3) Keith, J. A.; Vassilev-Galindo, V.; Cheng, B.; Chmiela, S.; Gastegger, M.; Müller, K.-R.; Tkatchenko, A. Combining Machine Learning and Computational Chemistry for Predictive Insights Into Chemical Systems. *Chemical Reviews* **2021**, *121*, 9816–9872.
- (4) Choudhary, K.; DeCost, B.; Chen, C.; Jain, A.; Tavazza, F.; Cohn, R.; Park, C. W.; Choudhary, A.; Agrawal, A.; Billinge, S. J., et al. Recent advances and applications of deep learning methods in materials science. *npj Computational Materials* **2022**, *8*, 59.
- (5) Xie, T.; Grossman, J. C. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Physical review letters* **2018**, *120*, 145301.

- (6) Karamad, M.; Magar, R.; Shi, Y.; Siahrostami, S.; Gates, I. D.; Farimani, A. B. Orbital graph convolutional neural network for material property prediction. *Physical Review Materials* **2020**, *4*, 093801.
- (7) Schütt, K. T.; Sauceda, H. E.; Kindermans, P.-J.; Tkatchenko, A.; Müller, K.-R. SchNet—A deep learning architecture for molecules and materials. *The Journal of Chemical Physics* **2018**, *148*, 241722.
- (8) Magar, R.; Wang, Y.; Lorsung, C.; Liang, C.; Ramasubramanian, H.; Li, P.; Farimani, A. B. AugLiChem: Data Augmentation Library of Chemical Structures for Machine Learning. *arXiv preprint arXiv:2111.15112* **2021**,
- (9) Magar, R.; Wang, Y.; Farimani, A. Crystal twins: self-supervised learning for crystalline material property prediction. *npj Comput. Mater* **2022**, *8*, 231.
- (10) Cao, Z.; Magar, R.; Wang, Y.; Barati Farimani, A. Moformer: self-supervised transformer model for metal–organic framework property prediction. *Journal of the American Chemical Society* **2023**, *145*, 2958–2967.
- (11) Louis, S.-Y.; Zhao, Y.; Nasiri, A.; Wang, X.; Song, Y.; Liu, F.; Hu, J. Graph convolutional neural networks with global attention for improved materials property prediction. *Physical Chemistry Chemical Physics* **2020**, *22*, 18141–18148.
- (12) Chen, C.; Ye, W.; Zuo, Y.; Zheng, C.; Ong, S. P. Graph networks as a universal machine learning framework for molecules and crystals. *Chemistry of Materials* **2019**, *31*, 3564–3572.
- (13) Choudhary, K.; DeCost, B. Atomistic Line Graph Neural Network for improved materials property predictions. *npj Computational Materials* **2021**, *7*, 1–8.
- (14) Ock, J.; Tian, T.; Kitchin, J.; Ulissi, Z. Beyond independent error assumptions in large GNN atomistic models. *The Journal of Chemical Physics* **2023**, *158*.

- (15) Kazeev, N.; Al-Maeeni, A. R.; Romanov, I.; Faleev, M.; Lukin, R.; Tormasov, A.; Castro Neto, A.; Novoselov, K. S.; Huang, P.; Ustyuzhanin, A. Sparse representation for machine learning the properties of defects in 2D materials. *npj Computational Materials* **2023**, *9*, 113.
- (16) Kim, S.; Noh, J.; Gu, G. H.; Aspuru-Guzik, A.; Jung, Y. Generative adversarial networks for crystal structure prediction. *ACS central science* **2020**, *6*, 1412–1420.
- (17) Xie, T.; Fu, X.; Ganea, O.-E.; Barzilay, R.; Jaakkola, T. S. Crystal Diffusion Variational Autoencoder for Periodic Material Generation. International Conference on Learning Representations. 2021.
- (18) Pollice, R.; dos Passos Gomes, G.; Aldeghi, M.; Hickman, R. J.; Krenn, M.; Lavigne, C.; Lindner-D'Addario, M.; Nigam, A.; Ser, C. T.; Yao, Z., et al. Data-driven strategies for accelerated materials design. *Accounts of Chemical Research* **2021**, *54*, 849–860.
- (19) Wang, S.; Guo, X.; Zhao, L. Deep generative model for periodic graphs. *arXiv preprint arXiv:2201.11932* **2022**,
- (20) Yao, Z.; Sánchez-Lengeling, B.; Bobbitt, N. S.; Bucior, B. J.; Kumar, S. G. H.; Collins, S. P.; Burns, T.; Woo, T. K.; Farha, O. K.; Snurr, R. Q., et al. Inverse design of nanoporous crystalline reticular materials with deep generative models. *Nature Machine Intelligence* **2021**, *3*, 76–86.
- (21) Ryan, K.; Lengyel, J.; Shatruk, M. Crystal structure prediction via deep learning. *Journal of the American Chemical Society* **2018**, *140*, 10158–10168.
- (22) Jain, A.; Ong, S. P.; Hautier, G.; Chen, W.; Richards, W. D.; Dacek, S.; Cholia, S.; Gunter, D.; Skinner, D.; Ceder, G.; Persson, K. A. Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. *APL Materials* **2013**, *1*, 011002.

- (23) Fung, V.; Zhang, J.; Juarez, E.; Sumpter, B. G. Benchmarking graph neural networks for materials chemistry. *npj Computational Materials* **2021**, *7*, 1–8.
- (24) Reiser, P.; Neubert, M.; Eberhard, A.; Torresi, L.; Zhou, C.; Shao, C.; Metni, H.; van Hoesel, C.; Schopmans, H.; Sommer, T., et al. Graph neural networks for materials science and chemistry. *Communications Materials* **2022**, *3*, 93.
- (25) Ruff, R.; Reiser, P.; Stühmer, J.; Friederich, P. Connectivity Optimized Nested Graph Networks for Crystal Structures. *arXiv preprint arXiv:2302.14102* **2023**,
- (26) Yan, K.; Liu, Y.; Lin, Y.; Ji, S. Periodic Graph Transformers for Crystal Material Property Prediction. *arXiv preprint arXiv:2209.11807* **2022**,
- (27) Lin, Y.; Yan, K.; Luo, Y.; Liu, Y.; Qian, X.; Ji, S. Efficient Approximations of Complete Interatomic Potentials for Crystal Property Prediction. *arXiv preprint arXiv:2306.10045* **2023**,
- (28) Gong, S.; Xie, T.; Shao-Horn, Y.; Gomez-Bombarelli, R.; Grossman, J. C. Examining graph neural networks for crystal structures: limitations and opportunities for capturing periodicity. *arXiv preprint arXiv:2208.05039* **2022**,
- (29) Goodall, R. E.; Parackal, A. S.; Faber, F. A.; Armiento, R.; Lee, A. A. Rapid discovery of stable materials by coordinate-free coarse graining. *Science Advances* **2022**, *8*, eabn4117.
- (30) Goodall, R. E.; Lee, A. A. Predicting materials properties without crystal structure: Deep representation learning from stoichiometry. *Nature Communications* **2020**, *11*, 1–9.
- (31) Ihalage, A.; Hao, Y. Formula Graph Self-Attention Network for Representation-Domain Independent Materials Discovery. *Advanced Science* **2022**, 2200164.

- (32) Chithrananda, S.; Grand, G.; Ramsundar, B. ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction. *arXiv preprint arXiv:2010.09885* **2020**,
- (33) Xu, C.; Wang, Y.; Farimani, A. B. TransPolymer: a Transformer-based Language Model for Polymer Property Predictions. *arXiv preprint arXiv:2209.01307* **2022**,
- (34) Elnaggar, A.; Heinzinger, M.; Dallago, C.; Rehwari, G.; Wang, Y.; Jones, L.; Gibbs, T.; Feher, T.; Angerer, C.; Steinegger, M., et al. ProtTrans: towards cracking the language of lifes code through self-supervised deep learning and high performance computing. *IEEE transactions on pattern analysis and machine intelligence* **2021**,
- (35) Lin, Z.; Akin, H.; Rao, R.; Hie, B.; Zhu, Z.; Lu, W.; Smetanin, N.; dos Santos Costa, A.; Fazel-Zarandi, M.; Sercu, T.; Candido, S., et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. *bioRxiv* **2022**,
- (36) Rao, R. M.; Liu, J.; Verkuil, R.; Meier, J.; Canny, J.; Abbeel, P.; Sercu, T.; Rives, A. MSA transformer. International Conference on Machine Learning. 2021; pp 8844–8856.
- (37) Yüksel, A.; Ulusoy, E.; Ünlü, A.; Doğan, T. SELFormer: Molecular Representation Learning via SELFIES Language Models. 2023.
- (38) Nijkamp, E.; Ruffolo, J.; Weinstein, E. N.; Naik, N.; Madani, A. ProGen2: Exploring the Boundaries of Protein Language Models. 2022.
- (39) Lipman, D. J.; Pearson, W. R. Rapid and sensitive protein similarity searches. *Science* **1985**, *227*, 1435–1441.
- (40) Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *Journal of chemical information and computer sciences* **1988**, *28*, 31–36.

- (41) Ramprasad, R.; Batra, R.; Pilania, G.; Mannodi-Kanakkithodi, A.; Kim, C. Machine learning in materials informatics: recent applications and prospects. *npj Computational Materials* **2017**, *3*, 54.
- (42) Wang, Y.; Magar, R.; Liang, C.; Barati Farimani, A. Improving Molecular Contrastive Learning via Faulty Negative Mitigation and Decomposed Fragment Contrast. *Journal of Chemical Information and Modeling* **2022**,
- (43) Wang, Y.; Wang, J.; Cao, Z.; Barati Farimani, A. Molecular contrastive learning of representations via graph neural networks. *Nature Machine Intelligence* **2022**, 1–9.
- (44) Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* **2019**,
- (45) Dunn, A.; Wang, Q.; Ganose, A.; Dopp, D.; Jain, A. Benchmarking materials property prediction methods: the Matbench test set and Automatminer reference algorithm. *npj Computational Materials* **2020**, *6*, 1–10.
- (46) Rosen, A. S.; Iyer, S. M.; Ray, D.; Yao, Z.; Aspuru-Guzik, A.; Gagliardi, L.; Notestein, J. M.; Snurr, R. Q. Machine learning the quantum-chemical properties of metal–organic frameworks for accelerated materials discovery. *Matter* **2021**, *4*, 1578–1597.
- (47) Wilmer, C. E.; Leaf, M.; Lee, C. Y.; Farha, O. K.; Hauser, B. G.; Hupp, J. T.; Snurr, R. Q. Large-scale screening of hypothetical metal–organic frameworks. *Nature chemistry* **2012**, *4*, 83–89.
- (48) Hiller, H. Crystallography and cohomology of groups. *The American Mathematical Monthly* **1986**, *93*, 765–779.

- (49) Wyckoff, R. W. G. *The Analytical Expression of the Results of the Theory of Space-groups*; Carnegie institution of Washington, 1922.
- (50) Bucior, B. J.; Rosen, A. S.; Haranczyk, M.; Yao, Z.; Ziebel, M. E.; Farha, O. K.; Hupp, J. T.; Siepmann, J. I.; Aspuru-Guzik, A.; Snurr, R. Q. Identification schemes for metal–organic frameworks to enable rapid search and cheminformatics analysis. *Crystal Growth & Design* **2019**, *19*, 6682–6697.
- (51) O’Keeffe, M.; Peskov, M. A.; Ramsden, S. J.; Yaghi, O. M. The reticular chemistry structure resource (RCSR) database of, and symbols for, crystal nets. *Accounts of chemical research* **2008**, *41*, 1782–1789.
- (52) Trepte, K.; Schwalbe, S. porE: A code for deterministic and systematic analyses of porosities. *Journal of Computational Chemistry* **2021**, *42*, 630–643.
- (53) Tshitoyan, V.; Dagdelen, J.; Weston, L.; Dunn, A.; Rong, Z.; Kononova, O.; Persson, K. A.; Ceder, G.; Jain, A. Unsupervised word embeddings capture latent knowledge from materials science literature. *Nature* **2019**, *571*, 95–98.
- (54) Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M., et al. Huggingface’s transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771* **2019**,
- (55) Choudhary, K.; Garrity, K. F.; Reid, A. C.; DeCost, B.; Biacchi, A. J.; Hight Walker, A. R.; Trautt, Z.; Hattrick-Simpers, J.; Kusne, A. G.; Centrone, A., et al. The joint automated repository for various integrated simulations (JARVIS) for data-driven materials design. *npj computational materials* **2020**, *6*, 173.
- (56) Downs, R. T.; Hall-Wallace, M. The American Mineralogist Crystal Structure Database. *American Mineralogist* **2003**, *88*, 247–250.

- (57) Chanussot, L.; Das, A.; Goyal, S.; Lavril, T.; Shuaibi, M.; Riviere, M.; Tran, K.; Heras-Domingo, J.; Ho, C.; Hu, W., et al. Open catalyst 2020 (OC20) dataset and community challenges. *Acs Catalysis* **2021**, *11*, 6059–6072.
- (58) Kim, C.; Huan, T. D.; Krishnan, S.; Ramprasad, R. A hybrid organic-inorganic perovskite dataset. *Scientific Data* **2017**, *4*, 1–11.
- (59) Elfwing, S.; Uchibe, E.; Doya, K. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks* **2018**, *107*, 3–11.
- (60) Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. International Conference on Learning Representations. 2019.
- (61) Loshchilov, I.; Hutter, F. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983* **2016**,
- (62) Choudhary, K.; Kalish, I.; Beams, R.; Tavazza, F. High-throughput identification and characterization of two-dimensional materials using density functional theory. *Scientific Reports* **2017**, *7*, 1–16.
- (63) Petretto, G.; Dwaraknath, S.; PC Miranda, H.; Winston, D.; Giantomassi, M.; Van Setten, M. J.; Gonze, X.; Persson, K. A.; Hautier, G.; Rignanese, G.-M. High-throughput density-functional perturbation theory phonons for inorganic materials. *Scientific data* **2018**, *5*, 1–12.
- (64) Petousis, I.; Mrdjenovich, D.; Ballouz, E.; Liu, M.; Winston, D.; Chen, W.; Graf, T.; Schladt, T. D.; Persson, K. A.; Prinz, F. B. High-throughput screening of inorganic compounds for the discovery of novel dielectric and optical materials. *Scientific data* **2017**, *4*, 1–12.
- (65) Ward, L.; Dunn, A.; Faghaninia, A.; Zimmermann, N. E.; Bajaj, S.; Wang, Q.; Mon-

- toya, J.; Chen, J.; Bystrom, K.; Dylla, M., et al. Matminer: An open source toolkit for materials data mining. *Computational Materials Science* **2018**, *152*, 60–69.
- (66) de Jong, M.; Chen, W.; Angsten, T.; Jain, A.; Notestine, R.; Gamst, A.; Sluiter, M.; Krishna Ande, C.; van der Zwaag, S.; Plata, J. J.; Toher, C.; Curtarolo, S.; Ceder, G.; Persson, K. A.; Asta, M. Charting the complete elastic properties of inorganic crystalline compounds. *Scientific Data* **2015**, *2*, 150009.
- (67) Castelli, I. E.; Olsen, T.; Datta, S.; Landis, D. D.; Dahl, S.; Thygesen, K. S.; Jacobsen, K. W. Computational screening of perovskite metal oxides for optimal solar light capture. *Energy Environ. Sci.* **2012**, *5*, 5814–5819.
- (68) Pham, T. L.; Kino, H.; Terakura, K.; Miyake, T.; Tsuda, K.; Takigawa, I.; Dam, H. C. Machine learning reveals orbital interaction in materials. *Science and Technology of Advanced Materials* **2017**, *18*, 756–765.
- (69) van der Maaten, L.; Hinton, G. Visualizing Data using t-SNE. *Journal of Machine Learning Research* **2008**, *9*, 2579–2605.

## Supplementary Information

### Pretraining Dataset

We used data from the JARVIS-Tools Database<sup>55</sup> for MatInFormer’s pretraining. This database includes the Open Catalyst dataset<sup>57</sup> and others. Details on these datasets and their sizes are in Table S1.

For pretraining dataset 1, we took data from five JARVIS-Tools datasets namely dft\_3d, qe\_tb, alignn\_ff\_db, hMOF<sup>47</sup> and cod.,<sup>56</sup> We combined all the materials from these datasets and removed any duplicate materials. When calculating lattice parameters with JARVIS tools,<sup>55</sup> we take the mean value of the repeat data. This was done because we need the lattice parameters for lattice parameter prediction(LPP) strategy.

Similarly, we developed pretraining dataset 2, by combining all the datasets from Table S1 and removing any duplicate materials. To explore the pretraining datasets in greater details we plot the crystal system distribution and the elemental distribution. We observe a difference between pretraining dataset 1 and pretraining dataset 2 both in terms of elemental and crystal system distribution. For the pretraining dataset 1 the crystal system and elemental distribution is shown in Figure S1 and Figure S2. For the pretraining dataset 2 the crystal system and elemental distribution is shown in Figure S3 and Figure S4.

Table S1: Pretraining datasets composition and Size. We use the large dataset repository JARVIS - Tools compile the pretraining datasets in this work.

Source	Dataset Name	Size
JARVIS - Tools	DFT3D	75993
JARVIS - Tools	QETB	829574
JARVIS - Tools	alignn ff db	307113
JARVIS - Tools	hMOF	137651
JARVIS - Tools	COD	431778
JARVIS - Tools	Open Catalyst	510214
	Pretraining Dataset 1	642459
	Pretraining Dataset 2	997528

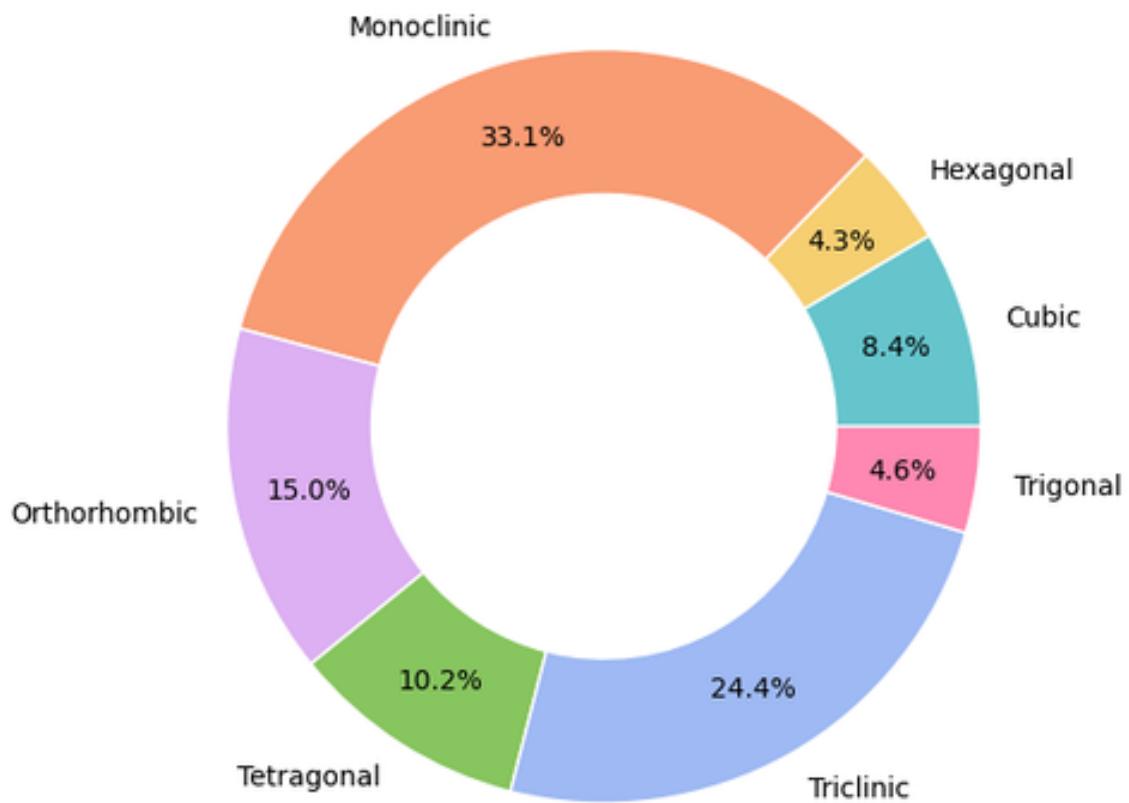


Figure S1: Crystal System distribution for Pretraining Dataset 1

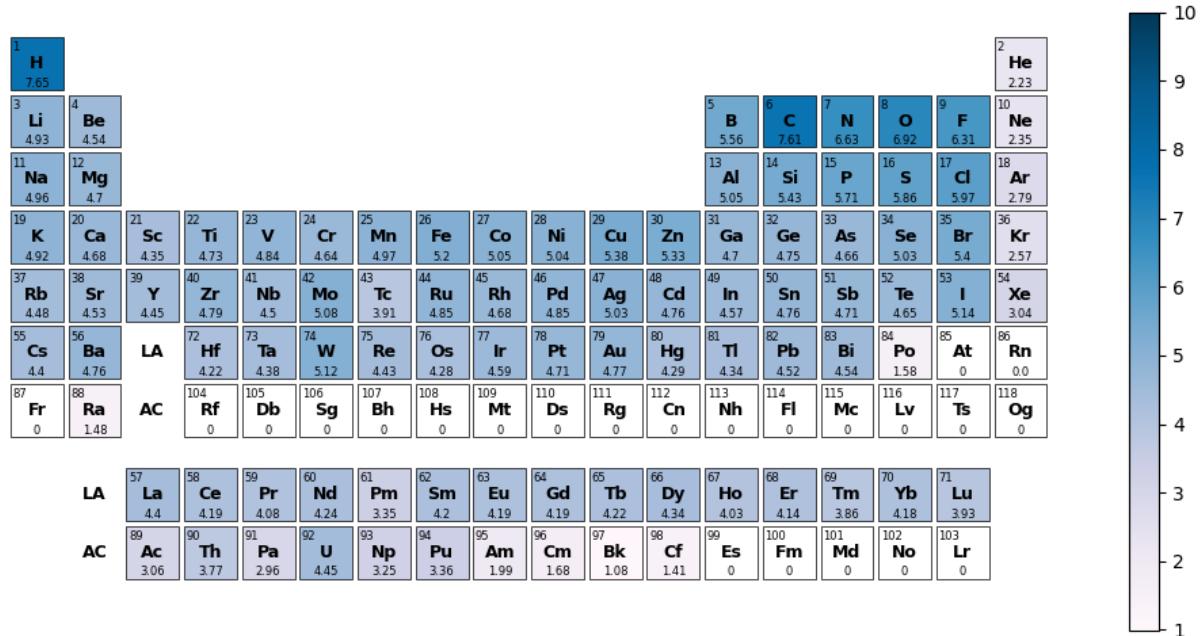


Figure S2: Element distribution for Pretraining Dataset 1

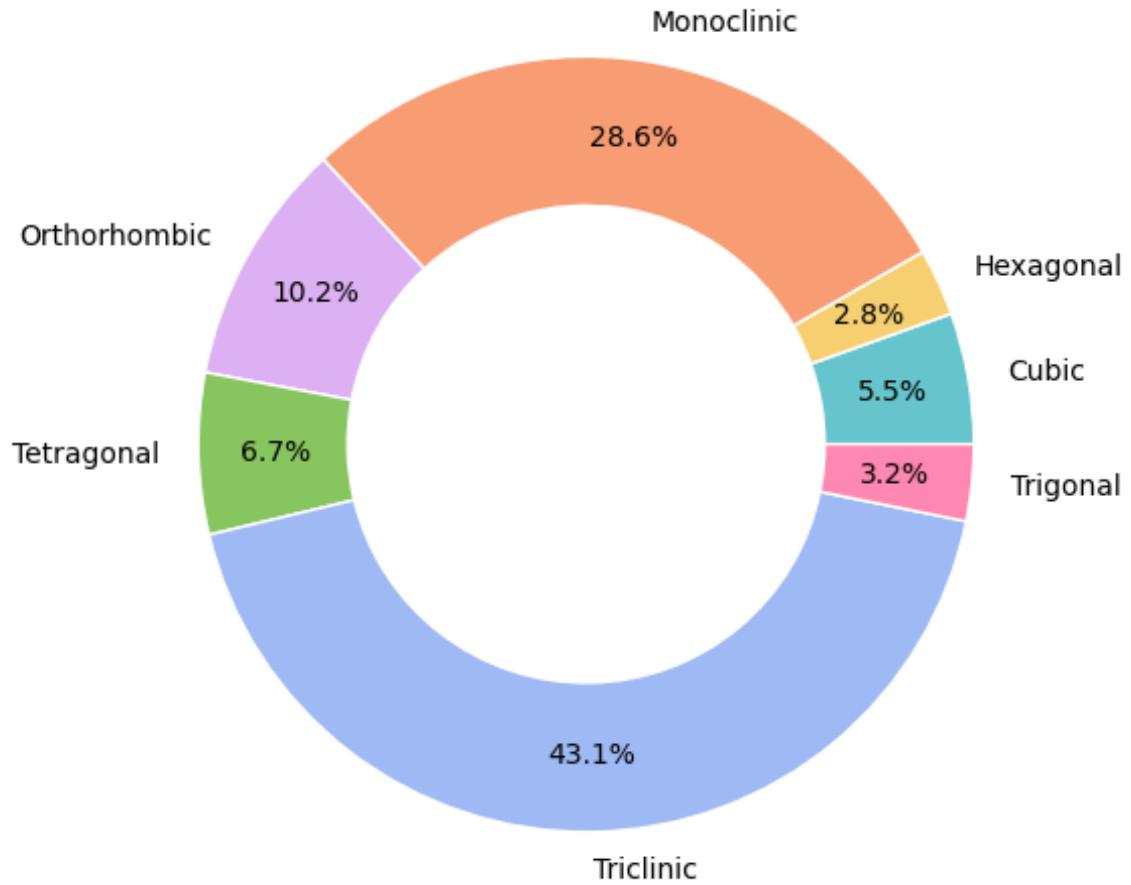


Figure S3: Crystal System distribution of Pretraining Dataset 2

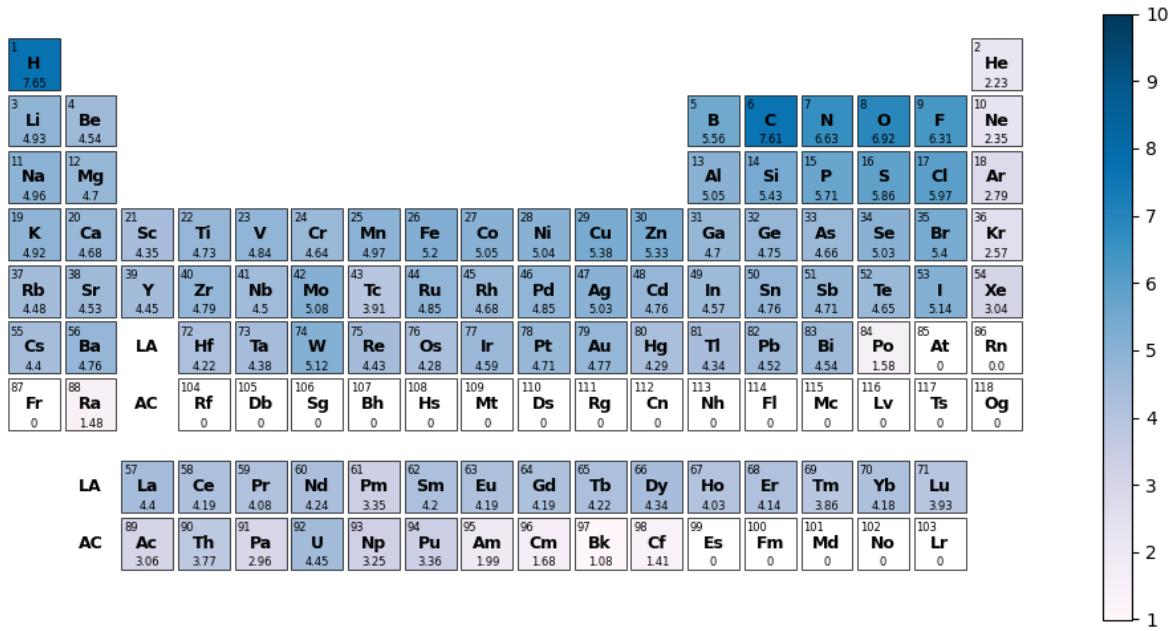


Figure S4: Element distribution of Pretraining Dataset 2

# Hyperparameters

## MatInFormer Hyperparameters

We used the Roberta<sup>44</sup> model as the architecture for the MatInFormer. The hyperparameters are detailed in Supplementary Table S2.

Table S2: Hyperparameters for the baseline Roberta Model which is used by the MatInFormer

Hyperparameters		Value
Layers		8
Heads number		12
Attention Dropout		0.1
Hidden Dropout		0.1
Linear Layer		2
Linear Dropout		0.1
Activation Function		SiLU
Hidden Size		768

## Pretraining Hyperparameters

Different pretraining strategies required various hyperparameters, as displayed in Supplementary Table S3. Given that Lattice Parameter Prediction (LPP) is more intricate than Masked Language Modeling (MLM), we used distinct hyperparameters for each. However, for the combined MLM + LPP strategy, we employed the same hyperparameters as those used for LPP.

We use the AdamW<sup>60</sup> optimizer with a learning rate of 1e-6 and weight decay of 0 for all pretraining strategies. The model was trained with a batch size of 64 for 200 epochs. A CosineAnnealingLR<sup>61</sup> scheduler with a warm-up is used to stabilize the training process. The learning rate was warmed up during the first 5 % of the total epoch and then decayed to zero as a cosine function for the rest of the epochs. The model occupied around 5GB memory and it takes 1 day to pretrain on an A6000 GPU.

Table S3: Pretrain Hyperparameter, LPP is lattice parameters prediction. LPP with mask is lattice parameters prediction with random masked space group tokens.

Method	Dataset	Epoch	Batch Size	Learning Rate	Weight Decay	Warm Up Ratio	Masking Ratio
MLM	642K	50	64	1e-6	0.0	0.05	0.25
LPP	997K	150	64	1e-6	0.0	0.05	0
LPP + MLM	642K	150	64	1e-6	0.0	0.05	0.25

## Finetuning Hyperparameters

For finetuning, the transformer encoder was loaded with the pretrained weights designed for downstream tasks. The hyperparameters used during this finetuning are detailed in Supplementary Table S4. For all models, we employed the AdamW<sup>60</sup> optimizer and the CosineAnnealingLR<sup>61</sup> scheduler. During pretraining, we utilized the Mean Square Error (MSE) loss for LPP and LPP + MLM, and cross-entropy for MLM. For the finetuning stage, the Mean Absolute Error (MAE) loss was employed. We would like to note that the hyperparameters during finetuning were optimized using a random search method.

Table S4: Finetuning Hyperparamter for the MatInFormer model. We used the same hyperparameter for finetuning all the different pretrained models. The hyperparameters for MLM if different have been indicated in (brackets) next to the same hyperparameter for the other finetuned models.

Dataset	Batch Size	Learning Rate	Weight Decay	Epoch
JDFT2D (JDFT) <sup>62</sup>	128	1e-5	0.001	50
Phonons <sup>63</sup>	128	1e-4	0.001	100
Dielectric <sup>64</sup>	128	1e-4	0.001	100
GVRH <sup>65,66</sup>	128	1e-4	0.01	200
KVRH <sup>66</sup>	128	1e-4	0.001	100
Perovskites <sup>67</sup>	128	1e-5	0.01	50
MP-Gap (MP-BG) <sup>22</sup>	512	1e-4	0.0001	200
MP-E-Form (MP-FE) <sup>22</sup>	128	1e-4	0.0001	200
hMOF <sup>47</sup>	128	1e-5	0.01	200

## Grid Point Approach for Porosity

We adapt the Grid Point Approach(GPA) from porE<sup>52</sup> for deterministic and systematic analyses of porosities in MOFs. Two properties are calculated, porosity fraction  $\Phi_{void}$  of the MOF structure and accessible porosity fraction  $\Phi_{acc}$ . The visual explanation is shown in Figure S5. The porosity is depends on the grid density and accesible porosity is related to the radius of the probe, for our experiment, we use the recommendation setting grid density  $\rho_{grid} = 5$  and radius of the probe is set to  $r_{probe} = 1.2$ .

$$\Phi_{void} = \frac{N_{unoccupied}}{N_{total}} \cdot 100\% \quad (\text{S1})$$

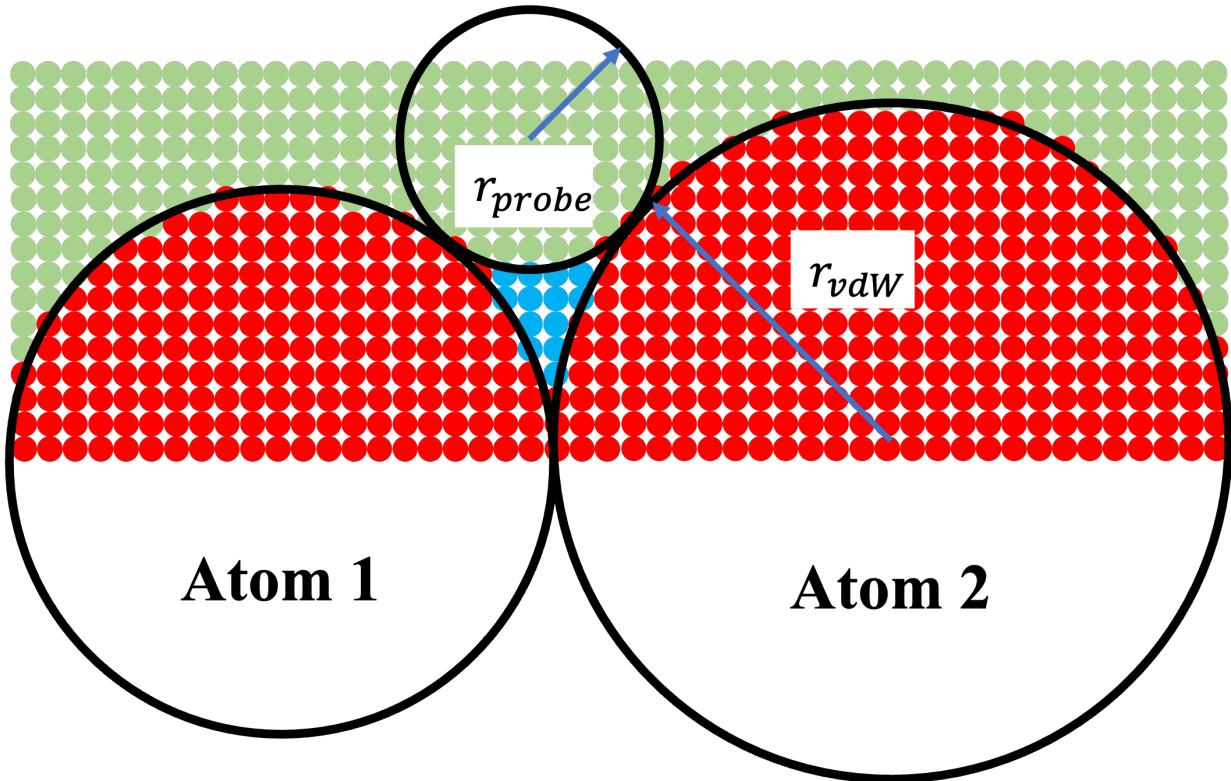


Figure S5: Visual explanation of different grid points in the grid point approach (GPA). Red points inside atoms vdW sphere are considered occupied. Green points are considered unoccupied, Blue points are unoccupied but unaccessible for certain values of  $r_{probe}$ <sup>52</sup>

## Hybrid Organic-Inorganic Perovskite(HOIP) result

In our study on the HOIP dataset, we adopted a 0.6/0.2/0.2 train/validation/test split ratio, with the results clearly outlined in Table S5. We incorporated 16 organic cations as informatics tokens in MatInFormer(O) and observed a marginal performance enhancement of 3.5%. This improvement is notably superior to that of structure-based GNNs, such as CGCNN. As highlighted in our results and discussion section, the position of atoms emerges as a significantly more crucial feature than the space group. Furthermore, a distinct advantage of our language model is its flexibility; it can easily integrate more informatics tokens. In contrast, GNNs encounter considerable challenges when attempting to encode additional features.

Table S5: Result of HOIP

Model	MAE
CGCNN	0.170 ±0.013
MatInFormer	0.336 ±0.002
MatInFormer(O)	0.324 ±0.001

## Attention Visualization of the CLS token

In Figure S6, we visualized the attention weights of the [CLS] tokens across different layers. We observe that attention weights focus on informatics tokens, such as topology, volume, and porosity. On the other hand there is negligible attention towards the space group tokens. This can be attributed to the fact that the majority of space groups for hMOF are "P1", making them less relevant for the model to distinguish between hMOFs. Instead, it is the informatics tokens, particularly topology, unit cell volume, and the number of atoms, that play a pivotal role in determining the property. This observation about attention weights aligns with our finetuning results presented in Table 4.

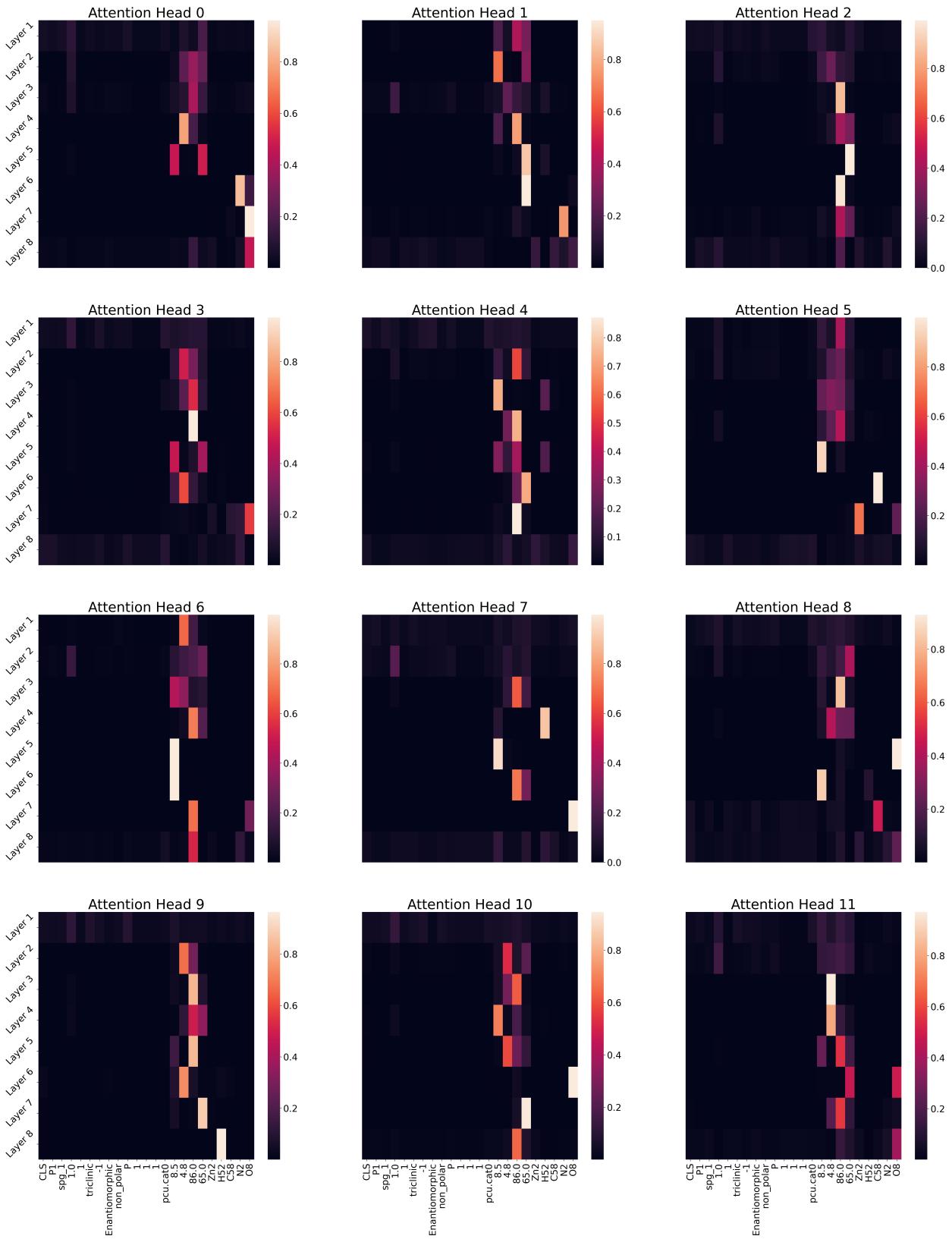


Figure S6: CLS Attention