ZATOM-1: Towards a Multimodal Foundation Model for 3D Molecules and Materials

Alex Morehead[1,2,*], **Miruna Cretu**[3,*], **Antonia Panescu**[4,*], **Rishabh Anand**[4,*], **Maurice Weiler**[5,*], **Tynan Perez**[5,*], **Samuel Blau**[1], **Steven Farrell**[1], **Wahid Bhimji**[1], **Anubhav Jain**[1], **Hrushikesh Sahasrabuddhe**[1,6], **Pietro Liò**[3], **Tommi Jaakkola**[5], **Rafael Gómez-Bombarelli**[5], **Rex Ying**[4,†], **N. Benjamin Erichson**[1,2,†], **Michael W. Mahoney**[6,1,2,†]

[1]LBNL [2]ICSI [3]University of Cambridge [4]Yale University [5]MIT [6]UC Berkeley
[*]Equal contribution [†]Equal advising

General-purpose 3D modeling in chemistry encompasses molecules and materials, requiring both generative and predictive capabilities. However, most existing AI approaches are optimized for a single domain (molecules or materials) and a single task (generation or prediction), which limits representation sharing and transfer. We introduce ZATOM-1, a cross-domain, general-purpose model architecture that unifies generative and predictive learning of 3D molecules and materials. ZATOM-1 is a deliberately simplified Transformer trained with a multimodal flow matching objective that jointly models discrete atom types and continuous 3D geometries. This approach supports scalable pretraining with predictable gains as model capacity increases, while enabling fast and stable sampling. We use cross-domain generative pretraining as a universal initialization for downstream multi-task prediction of properties, energies, and forces. Empirically, ZATOM-1 outperforms or competes with specialized baselines on both multi-task generative and predictive benchmarks in data-controlled settings, while improving generative inference speed by more than an order of magnitude. Our experiments demonstrate positive predictive transfer between data domains from joint generative pretraining: modeling materials during generative pretraining improves molecular property prediction accuracy. Open-source code and model weights are freely available.

Correspondence: AM: acmwhb@lbl.gov; MWM: mmahoney@stat.berkeley.edu
Code: https://github.com/Zatom-AI/zatom
Keywords: 3D Molecules, Materials Design, Foundation Models, Multimodal Flow Matching

1 Introduction

Life, materials, and many modern functions, ranging from small molecule therapeutics to consumer electronic devices, are driven by chemistry. However, navigating the design space of chemical systems for high-impact downstream applications is notoriously time-consuming and error-prone [7, 101, 105]. As such, developing our understanding of chemical systems with artificial intelligence (AI) methods, in particular foundation models equipped with neural scaling [8, 51, 30], represents a promising area of investment for deep learning research, given that related computational techniques have recently led to significant breakthroughs across adjacent scientific fields [50, 1, 37].

Recently, deep learning has begun to unify the generation of new 3D molecules and materials [18, 49, 121, 67], and several specialized prediction models currently exist for advanced molecule and material property prediction [62, 80, 78]. Nonetheless, no prior works offer a general-purpose model capable of both generative modeling and representation learning of 3D molecules and materials via cross-domain pretraining. To build such a unified model, several design choices must be addressed, including how to tokenize its inputs as well as what and how to learn using these tokens.

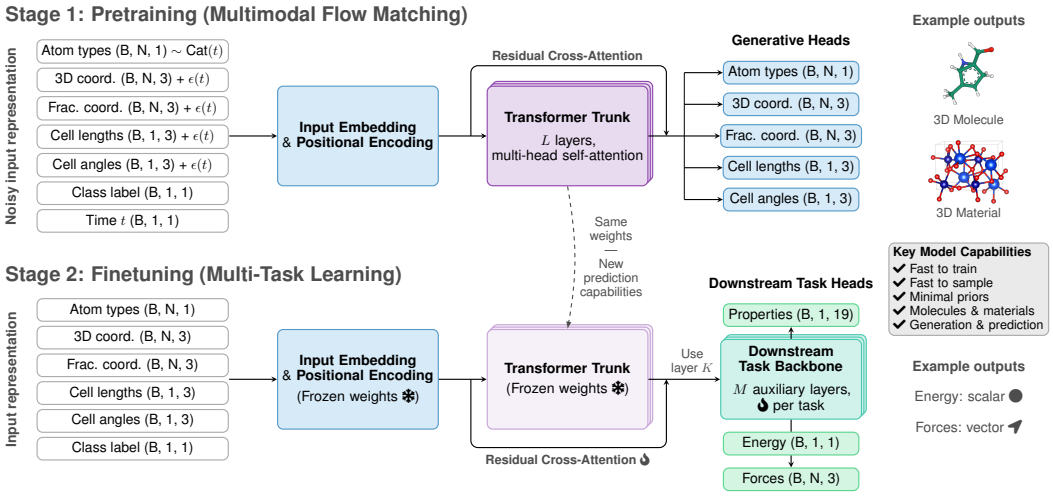


Figure 1: **The ZATOM-1 architecture for 3D molecules and materials.** A unified Transformer trunk processes (noisy) atomic, structural, and conditional inputs. The model features two training stages: (1) multimodal flow pretraining using the final trunk layer L's representations to create new molecules or materials; and (2) multi-task finetuning of additional downstream task backbones and heads that use a specified trunk layer K's representations to predict properties, energies, and forces.

Inspired by previous research on scientific foundation models [94, 1] and neural scaling [95], we argue for the importance of tokenizing atomistic systems at the fundamental level of atomic granularity: 3D atom coordinates and element types. The next question is what to learn with these tokens. Previous works have shown that image and video diffusion models, employing standardized deep learning architectures, can learn rich embeddings for downstream prediction tasks in a self-supervised manner [61, 99] without assuming a canonical token ordering, as is common in autoregressive models [17]. In addition, scientific machine learning (SciML) is increasingly adopting standardized deep learning architectures to train highly scalable models [1, 54, 104]. As such, it is clear that non-autoregressive generative modeling, due to its self-supervised (i.e., label-free) nature, has the potential to serve as a powerful and flexible pretraining method for standardized atomistic model architectures.

In this spirit, we propose a question: *How much chemical knowledge can we represent with a 3D generative model?* **ZATOM-1**, our answer to this question, is illustrated in Figure 1. It is a general-purpose, multimodal flow architecture for 3D molecules and materials, based on these ideas:

1. *Ambient all-atom generative modeling.* We represent 3D molecules and materials in $\mathbb{R}^3$ and perform generative flow matching directly within this Euclidean space. Importantly, with this approach, no pretrained autoencoders are necessary, e.g., for latent diffusion [49], delivering notable speed-ups in model training and inference.
2. *Representation learning using generative models.* We frame generative modeling of 3D molecules and materials as an ideal pretraining task for chemical representation learning [10] using modern, standardized Transformers [98].

Contributions. Based on these ideas, our main contributions in this work are as follows:

- We introduce ZATOM-1, a general-purpose model architecture for diverse 3D chemical tasks, including *end-to-end* generation of periodic 3D materials and non-periodic 3D molecules, multi-task (and for materials, zero-shot) property prediction, and energy and force modeling.
- ZATOM-1: (**1**) achieves *state-of-the-art* for 3D materials (molecule) generation with LeMat-GenBench (QM9 & GEOM-Drugs); (**2**) provides state-of-the-art multi-task QM9 property predictions; (**3**) yields a 3x reduction in GPU training hours compared to latent diffusion [49]; and (**4**) demonstrates positive transfer learning through cross-domain pretraining.
- The ZATOM-1 architecture features a standardized Transformer with Query-Key Normalization [110], Flash Attention [22], and SwiGLU feed-forward networks [86], yielding predictably improving performance through parameter scaling. Its design enables a **12.5x**

speed-up over a 500M-parameter latent diffusion baseline [49], with ZATOM-1 (300M) generating 10,000 samples in under 4 minutes on a single NVIDIA A100 GPU.

2 ZATOM-1

In this section, we introduce the ZATOM-1 model architecture for generative modeling and representation learning of 3D molecules and materials. Namely, we describe its unified input representation and architectural features for such data types and characterize how we pretrain and finetune ZATOM-1 for a variety of tasks in 3D chemistry.

In particular, ZATOM-1 is trained in two stages: (1) generative pretraining for 3D molecule and material generation; and (2) predictive finetuning for energy, force, and property prediction tasks. For generative pretraining, we use conditional flow matching, a scalable training technique for normalizing flows [64]. As a *non-autoregressive* generative modeling paradigm [73, 74], conditional flow matching trains a neural network to approximate a time-conditioned velocity field required to transport samples from a source distribution to a target distribution using ordinary differential equation (ODE) integration. Importantly, time integration is only required during sampling, not model training, offering scalable training speeds as a result [71].

To efficiently model the multimodal nature of 3D atomic systems, we use multimodal flow matching over a (discrete) atom type modality and four (continuous) geometric modalities, yielding a joint distribution of these modalities for generative modeling. Previous works for 3D materials generation have attempted multimodal flow matching over similar modality structures [68]. However, we argue that reliance on sparse graph neural network architectures, as well as hand-crafted generative priors, has hindered performance and training/inference speed for generative modeling in this domain [55, 80]. Instead, we adopt a standardized Transformer architecture and Gaussian multimodal flow matching for scalable joint modeling of 3D molecules and materials. We discuss related works and ZATOM-1's placement in the literature in detail in Appendix A.

2.1 Problem formulation

We aim to learn a joint generative model of 3D molecules and materials and to obtain rich neural embeddings of both data domains. Towards this end, we represent a 3D molecule or material with N atoms using five unified modalities:

Unified Input Representation

$\boldsymbol{A} = \{a_i\}_{i=1}^N \in \mathbb{Z}^{1\times N}$	(Atom types)	$\boldsymbol{L}_{\text{len}} = \{l_{\text{len}}\} \in \mathbb{R}^{3\times 1}$	(Lattice lengths)
$\boldsymbol{X} = \{x_i\}_{i=1}^N \in \mathbb{R}^{3\times N}$	(3D coordinates)	$\boldsymbol{L}_{\text{ang}} = \{l_{\text{ang}}\} \in \mathbb{R}^{3\times 1}$	(Lattice angles)
$\boldsymbol{F} = \{f_i\}_{i=1}^N \in [0,1)^{3\times N}$	(Fractional coordinates)		

Atom types $\boldsymbol{A}$ are a discrete modality defined over the set of integers $\mathbb{Z}$, while 3D coordinates $\boldsymbol{X}$ are a continuous modality of $\mathbb{R}^3$ expressed in Angstroms. Fractional coordinates $\boldsymbol{F} = \boldsymbol{L}^{-1}\boldsymbol{X}$ reside in the range $[0, 1)$, and rotation-invariant lattice side lengths $\{l_{\text{len}}\} = \{a, b, c\} \in \mathbb{R}^{3\times 1}$ and the three internal angles between them $\{l_{\text{ang}}\} = \{\alpha, \beta, \gamma\} \in [60^\circ, 120^\circ]^{3\times 1}$ parametrize a parallelepiped $\boldsymbol{L} \in \mathbb{R}^{3\times 3}$ [35] characterizing the repeating unit cell [68]. For the sake of numerical stability, during training and sampling, lattice lengths are normalized by $\sqrt[3]{N}$, and lattice angles are converted from degrees to radians. Further, during training and sampling, for non-periodic molecules the lattice lengths, angles, and fractional coordinate inputs are masked to null values $\varnothing$. For periodic materials 3D coordinate inputs are masked to null values $\varnothing$ to enforce materials rotation invariance by default (with training losses masked accordingly).

2.2 Network architecture

We use a standard Transformer design, titled the **Trunk-based Flow Transformer (TFT)**, to learn a shared latent representation of each molecular and material modality. For a given 3D atomic system $(\boldsymbol{A}, \boldsymbol{X}, \boldsymbol{F}, \boldsymbol{L}_{\text{len}}, \boldsymbol{L}_{\text{ang}})$ with a periodic (material) or non-periodic (molecule) class label c, denoising timestep index t, and (optional) additive noise, a TFT model $\mathcal{T}$ with L Transformer trunk layers learns latent representations $\boldsymbol{Z} = \{z_i\}_{i=1}^N \in \mathbb{R}^{d\times N}$ from each molecular and material modality to predict denoised modalities $\boldsymbol{S} = \left(\boldsymbol{A}', \boldsymbol{X}', \boldsymbol{F}', \boldsymbol{L}'_{\text{len}}, \boldsymbol{L}'_{\text{ang}}\right)$ and auxiliary properties

Algorithm 1 Pseudocode for TFT model $\mathcal{T}$

Input: Noisy 3D atoms $(\{a_i\}, \{x_i\}, \{f_i\}, l_{\text{len}}, l_{\text{ang}})$, class c, time t
Output: Denoised 3D atoms $(\{a'_i\}, \{x'_i\}, \{f'_i\}, l'_{\text{len}}, l'_{\text{ang}})$, predictions $\{p^{\text{props}'}, p^{\text{energy}'}, p_i^{\text{forces}'}\}$

Lin^{B}, Lin^{NB} = Linear(bias=True), Linear(bias=False)
Mask inputs and project to d_{model}
x_i, $(f_i, l_{\text{len}}, l_{\text{ang}}) = x_i \cdot \mathbb{I}_{\text{molecule}}$, $(f_i, l_{\text{len}}, l_{\text{ang}}) \cdot \mathbb{I}_{\text{material}}$
$h_i = \text{Embed}(a_i) + \text{Embed}(c) + \text{Embed}(t) + \sum_{v \in \{x_i, f_i, l_{\text{len}}, l_{\text{ang}}\}} \text{Lin}_v^{\text{NB}}(v)$ $\quad h_i \in \mathbb{R}^{d_{\text{model}}}$
Apply encoder trunk (L layer transformer)
$\{z_i, z_i^{\text{K}}\} = \text{TransformerEncoder}^{L}(\{h_i\})$
Apply residual cross-attention to input embeddings
$\{h_i^a, h_i^x, h_i^f, h_i^{l_{\text{len}}}, h_i^{l_{\text{ang}}}\} = \text{TransformerDecoderBlock}(\{z_i, h_i\})$
Denoise inputs
$a'_i = \text{argmax}(\text{Lin}^{\text{B}}(\text{SiLU}(\text{Lin}^{\text{B}}(h_i^a))))$ $\quad a'_i \in \mathbb{Z}$
$x'_i = \text{Lin}^{\text{NB}}(\text{LayerNorm}(h_i^x))$ $\quad x'_i \in \mathbb{R}^3$
$f'_i = \text{Lin}^{\text{NB}}(\text{LayerNorm}(h_i^f))$ $\quad f'_i \in \mathbb{R}^3$
$l'_{\text{len}} = \text{Lin}^{\text{NB}}\left(\text{LayerNorm}\left(\frac{1}{N}\sum_{i=1}^{N} h_i^{l_{\text{len}}}\right)\right)$ $\quad l' \in \mathbb{R}^3$
$l'_{\text{ang}} = \text{Lin}^{\text{NB}}\left(\text{LayerNorm}\left(\frac{1}{N}\sum_{i=1}^{N} h_i^{l_{\text{ang}}}\right)\right)$ $\quad l' \in \mathbb{R}^3$
Predict outputs (cross-attention + M layer transformer)
$\{h_i^{\text{props}}, h_i^{\text{energy}}, h_i^{\text{forces}}\} = \text{TransformerDecoderBlock}_y(\{z_i^{\text{K}}, h_i\})$, for $y \in \{\text{props, energy, forces}\}$
Update $\{h_i^y\} \leftarrow \text{TransformerEncoder}_y^{M}(\{h_i^y\})$
$p^{\text{props}'} = \text{Lin}^{\text{B}}\left(\frac{1}{N}\sum_{i=1}^{N} h_i^{\text{props}}\right)$ $\quad p \in \mathbb{R}^{19}$
$p^{\text{energy}'} = \text{Lin}^{\text{B}}\left(\frac{1}{N}\sum_{i=1}^{N} h_i^{\text{energy}}\right)$ $\quad p \in \mathbb{R}^{1}$
$p_i^{\text{forces}'} = \text{Lin}^{\text{NB}}\left(h_i^{\text{forces}}\right)$ $\quad p \in \mathbb{R}^{3}$

Algorithm 2 Pseudocode for multimodal sampling

Input: Denoising model $\mathcal{T}$, number of atoms N, sampling steps $\{t_i\}_{i=0}^{S}$
Output: Sampled 3D atoms $(\mathbf{a}, \mathbf{x}, \mathbf{f}, \mathbf{l}_{\text{len}}, \mathbf{l}_{\text{ang}})$

def DiscreteFlowStep($\mathbf{a}_t, \mathbf{p}'_{t,a}, t, \Delta t$):
  # NOTE: All indexed ops are vectorized
  $\mathbf{r}_t(i, \cdot) \leftarrow \frac{\Delta t}{1-t}\mathbf{p}'_{t,a}(i)$ [13]
  $\mathbf{r}_t(i, \mathbf{a}_t(i)) \leftarrow -\sum_{j \neq \mathbf{a}_t(i)} \mathbf{r}_t(i, j)$
  $\mathbf{p}_{t+\Delta t}(i, j) \leftarrow 1_{\mathbf{a}_t(i)=j} + \mathbf{r}_t(i, j)$
  $\mathbf{a}_t(i) \leftarrow \text{Categorical}(\mathbf{p}_{t+\Delta t}(i, \cdot))$
  return $\mathbf{a}_t$

def EuclideanStep($\mathbf{z}_t, \mathbf{z}'_t, t, \Delta t, g(\cdot), \gamma_g$):
  $\mathbf{v}_t \leftarrow \frac{1}{1-t}(\mathbf{z}'_t - \mathbf{z}_t)$ [34]
  $\mathbf{s}_t \leftarrow g(t)\frac{t\mathbf{v}_t - \mathbf{z}_t}{1-t}$
  $d\mathbf{W}_t \leftarrow \sqrt{2\gamma_g\, g(t)}\,\mathcal{N}(0, I)$
  $\mathbf{z}_t \leftarrow (\mathbf{v}_t + \mathbf{s}_t + d\mathbf{W}_t)\Delta t$
  return $\mathbf{z}_t$

Initialize all modalities with random noise
$\mathbf{a}_t \leftarrow \text{Categorical}(\delta(\frac{1}{\#\text{ atom types}}))$
Initialize $\mathbf{z}_t \leftarrow \mathcal{N}(0, I)$ for each continuous modality $\mathbf{z} \in \{\mathbf{x}, \mathbf{f}, \mathbf{l}_{\text{len}}, \mathbf{l}_{\text{ang}}\}$
# Iteratively denoise over sampling steps
for $i = 1$ to S **do**
  $\Delta t \leftarrow t_i - t_{i-1}$
  # Predict denoised endpoints using the model $\mathcal{T}$
  $(\mathbf{p}'_{t,a}, \mathbf{x}'_t, \mathbf{f}'_t, \mathbf{l}'_{t,\text{len}}, \mathbf{l}'_{t,\text{ang}}) \leftarrow \mathcal{T}(\mathbf{a}_t, \mathbf{x}_t, \mathbf{f}_t, \mathbf{l}_{t,\text{len}}, \mathbf{l}_{t,\text{ang}}, t_i)$
  # Perform one sampling step for each modality
  $\mathbf{a}_t \leftarrow \text{DiscreteFlowStep}(\mathbf{a}_t, \mathbf{p}'_{t,a}, t_i, \Delta t)$
  Update $\mathbf{z}_t \leftarrow \text{EuclideanStep}(\mathbf{z}_t, \mathbf{z}'_t, t_i, \Delta t)$ for each $\mathbf{z}$
end for
# Return final denoised sample
return $(\mathbf{a}, \mathbf{x}, \mathbf{f}, \mathbf{l}_{\text{len}}, \mathbf{l}_{\text{ang}})$

$$\boldsymbol{P} = \left(\boldsymbol{P}^{\text{props}'}, \boldsymbol{P}^{\text{energy}'}, \boldsymbol{P}^{\text{forces}'}\right) = \left(p^{\text{props}'} \in \mathbb{R}^{19},\ p^{\text{energy}'} \in \mathbb{R}^{1},\ \{p_i^{\text{forces}'}\}_{i=1}^{N} \in \mathbb{R}^{3\times N}\right):$$
$$\boldsymbol{S}, \boldsymbol{P} \ = \ \mathcal{T}\left(\boldsymbol{A}, \boldsymbol{X}, \boldsymbol{F}, \boldsymbol{L}_{\text{len}}, \boldsymbol{L}_{\text{ang}}, c, t\right), \tag{1}$$

where a 10% drop probability is applied to c during training to support (optional) classifier-free guidance during inference, and additional (e.g., property) conditioning can be added likewise.

Algorithm 1 provides pseudocode for the forward pass of a TFT model with L trunk layers, where layer $K \leq L$ is used for auxiliary task representations. Using random data augmentations during training, the model learns the rotational and periodic symmetries of molecules and materials, respectively [1, 49]. Additionally, inspired by the Platonic Transformer [44], we experiment with an O(3)-equivariant version of ZATOM-1 (PLATOM-1) featuring a TFT model variant titled the **Trunk-based Flow Platoformer (TFP)**, which we describe in more detail in Appendix E.

2.3 Stage 1: Multimodal flow pretraining

Conceptually, given a pair of data samples $(\mathbf{x}_0, \mathbf{x}_1)$ from their respective source and target distributions, flow matching (FM) offers a scalable technique for training generative models. Namely, using FM, one can train a neural network $v_\theta(\mathbf{x}_t, t)$ to sample data from the target distribution by matching the expected velocities transporting samples from the (often simpler) source to the target distribution:

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{t,(\mathbf{x}_0,\mathbf{x}_1)}[\|v_\theta(\mathbf{x}_t, t) - u_t\|_2^2], \tag{2}$$

where $\mathbf{x}_t = (1-t)\mathbf{x}_0 + t\mathbf{x}_1$ and $u_t = \mathbf{x}_1 - \mathbf{x}_0$. In this work, to optimize continuous modalities, we use Euclidean diffusion/conditional flow matching (CFM) [88, 89, 39, 64, 2, 97], leveraging the equivalence between these two methods [32]. For optimizing atom types, we employ Discrete CFM, which is parameterized within the framework of Discrete Flow Models [13].

Now, specifically consider a molecule or material comprised of N atoms, characterized by zero-centered ground-truth 3D coordinates $\boldsymbol{X}$, fractional coordinates $\boldsymbol{F}$, lattice lengths $\boldsymbol{L}_{\text{len}}$, angles $\boldsymbol{L}_{\text{ang}}$, and atom types $\boldsymbol{A}$. The continuous 3D coordinates and fractional coordinates, along with lattice lengths and angles, undergo partial noise perturbation represented as $\boldsymbol{X}_t = t \cdot \boldsymbol{X} + (1-t) \cdot \epsilon$, where the noise ϵ is drawn from a Gaussian distribution $\mathcal{N}(0, I)$. For the discrete modality, the noisy atom types $\boldsymbol{A}_t$ are derived by interpolating between atom type probabilities and sampling from a categorical distribution, expressed as $\boldsymbol{A}_t \sim \text{Cat}(t \cdot \delta(\boldsymbol{A}) + (1-t) \cdot \delta(\frac{1}{\#\text{ atom types}}))$, where $\delta(\cdot)$ generates a one-hot encoding [13]. During the training phase, the model receives $\left(\boldsymbol{X}_t, \boldsymbol{F}_t, \boldsymbol{L}_{t_{\text{len}}}, \boldsymbol{L}_{t_{\text{ang}}}, \boldsymbol{A}_t\right)$ as input and learns to predict the terminal points of the trajectory.

Flow matching losses. The objective functions for each continuous and discrete modality, respectively, are defined as follows:

$$\mathcal{L}_{\text{metric}}(\boldsymbol{X}) = \mathbb{E}_{\epsilon,t}\left[\frac{1}{N}\|\boldsymbol{X}' - \boldsymbol{X}\|_2^2\right] \qquad \mathcal{L}_{\text{discrete}}(\boldsymbol{A}) = \mathbb{E}_t\left[-\sum_i a_i \log(a_i')\right] \tag{3}$$

These objectives are consolidated into a multi-objective framework, employing a weighting factor $\lambda_{\text{discrete}} \in [0, 1]$, expressed as

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{metric}}(\boldsymbol{X}) + \mathcal{L}_{\text{metric}}(\boldsymbol{F}) + \mathcal{L}_{\text{metric}}(\boldsymbol{L}_{\text{len}}) + \mathcal{L}_{\text{metric}}(\boldsymbol{L}_{\text{ang}}) + \lambda_{\text{discrete}} \cdot \mathcal{L}_{\text{discrete}}(\boldsymbol{A}). \tag{4}$$

During training, we set $\lambda_{\text{discrete}} = 0.1$ and sample from $t \sim \text{Beta}(\alpha_t, 1)$, where $\alpha_t = 1.8$, in line with prior studies [34, 103]. Further, to learn data symmetries and accelerate convergence during training, we randomly rotate and (for periodic materials) translate multiple copies of each input example, which are noised to different levels according to batch-wise t [1]. As $t \to 1$, the model resembles an identity function, since we adopt an endpoint formulation for flow matching due to its empirical performance for scientific applications [92]. To enable the model to effectively learn accurate atom placements even when $t \to 1$ causes the losses to approach zero, based on the sampled time t, we scale the loss with $\beta(t) \cdot \mathcal{L}_{\text{total}}(\boldsymbol{X}_t, \boldsymbol{F}_t, \boldsymbol{L}_{t_{\text{len}}}, \boldsymbol{L}_{t_{\text{ang}}}, \boldsymbol{A}_t)$, where $\beta(t) = \min\left\{100,\ \frac{1}{(1-t)^2}\right\}$.

Multimodal sampling. Algorithm 2 describes how we perform unconditional sampling of each continuous and discrete modality to generate 3D molecules and materials. Concisely, we sample random atom types or Gaussian noise for discrete and continuous modalities, respectively comprising N atoms (where N is sampled according to training dataset statistics [41]), and iteratively denoise each modality using a series of Euler steps operating on the multimodal predictions of the TFT model $\mathcal{T}$. Notably, other ODE solvers can be used, though we find that multimodal Euler steps provide strong empirical performance without considerable hyperparameter tuning.

2.4 Stage 2: Finetuning

As illustrated in Figure 1, we treat conditional flow matching of the modalities $(\boldsymbol{A}, \boldsymbol{X}, \boldsymbol{F}, \boldsymbol{L}_{\text{len}}, \boldsymbol{L}_{\text{ang}})$ as a self-supervised pretraining task akin to bidirectional modeling used to pretrain non-autoregressive BERT-style models [25] for enhanced representation learning [122] and *label-free* transfer learning. Once the model weights listed up to Line 10 of Algorithm 1 have been (pre)trained, for finetuning, (based on ablations described in Table 14 of Appendix F) we freeze these weights and enable training of the M-layer auxiliary Transformer trunk weights following Line 10 using trunk layer K's frozen embeddings. By default, $K = L$, a choice that we ablate in Table 4 of Section 4.

Predictive Losses. For property prediction, we supervise the task's embeddings $p_b^{\text{props'}}$ using the mean absolute error (MAE) over each batch element b as follows:

$$\mathcal{L}_{\text{MAE}}(\boldsymbol{P}^{\text{props'}}) = \frac{1}{B}\sum\nolimits_b \left|p_b^{\text{props'}} - p_b^{\text{props}}\right|, \tag{5}$$

where B is the number of samples in a given mini-batch, $p_b^{\text{props'}}$ denotes the predicted property (or properties) for the b-th sample, and p_b^{props} denotes the true property for the same sample. Similarly, energy embeddings $p_b^{\text{energy'}}$ and force embeddings $p_b^{\text{forces'}}$ are supervised using the batch-wise mean squared error (MSE), with the loss weight λ_{forces} specifically applied for forces:

$$\mathcal{L}_{\text{MSE}}(\boldsymbol{P}^{\text{energy'}}) = \frac{1}{B}\sum\nolimits_b \left(p_b^{\text{energy'}} - p_b^{\text{energy}}\right)^2 \quad \mathcal{L}_{\text{MSE}}(\boldsymbol{P}^{\text{forces'}}) = \mathbb{E}\left[\frac{1}{N}\|\boldsymbol{P}^{\text{forces'}} - \boldsymbol{P}^{\text{forces}}\|_2^2\right] \tag{6}$$

3 Experimental Setup

In this section, we describe how we train and evaluate ZATOM-1's performance for generative modeling and representation learning of 3D molecules and materials. Additionally, we discuss the representative baselines against which we comprehensively compare ZATOM-1.

Datasets. To fairly compare ZATOM-1 to previous methods [49], we train models on periodic materials from MP20 and non-periodic molecules from QM9 for our baseline generative pretraining

Table 1: **Crystal generation results on LeMat-GenBench with MP20 training.** Core model evaluation metrics are calculated for 2,500 sampled crystals using an *MLIP ensemble* with *LeMat-Bulk* as the reference set. Best values are shown in **bold**, second-best values are underlined. ⋆ denotes aggregate metrics that, summed together, constitute the primary objective to optimize for this benchmark (i.e., SUN + MSUN). Notably, jointly trained ZATOM-1 (80M) achieves its results using ∼400 GPU training hours, 3x faster than jointly trained ADiT (180M). Further, jointly trained ZATOM-1-WD achieves state-of-the-art SUN (and MSUN) rates for materials discovery in both relaxed (and unrelaxed) method categories [24] through aggressive weight decay ($1e^{-2}$) and early stopping ($1/5$ into training, before molecule validation performance has fully converged).

Model	# Params	Valid% ↑	Unique% ↑	Novel% ↑	Energy-based			Stability		Metastability	
					E_f (Std) ↓	E_{hull} (Std) ↓	RMSD (Std) ↓	Stable% ↑	⋆ SUN% ↑	Metastable% ↑	⋆ MSUN% ↑
With relaxation											
Crystalite	67M	**97.2**	**95.8**	53.2	−0.89	0.09	0.13	**12.7**	1.5 (38)	51.6	22.6
MatterGen	47M	95.7	95.1	70.5	-0.70 ± 0.79	0.18 ± 0.18	0.39 ± 0.50	2.0	0.2 (6)	33.4	15.0
PLaID++	7B	96.0	77.8	24.2	-0.50 ± 0.44	0.09 ± 0.16	$\mathbf{0.13 \pm 0.29}$	12.4	1.0 (26)	60.7	7.6
WyFormer	8B	93.4	93.0	66.4	-0.43 ± 0.95	0.50 ± 0.51	0.81 ± 0.98	0.5	0.1 (3)	15.7	1.9
WyFormer-DFT	8B	95.2	95.0	66.4	-0.67 ± 0.91	0.27 ± 0.36	0.42 ± 0.60	3.7	0.4 (9)	24.8	7.8
Without relaxation											
Jointly trained ADiT	180M	90.6	87.8	26.0	-0.73 ± 0.93	0.33 ± 0.45	0.38 ± 0.40	0.4	0.0 (0)	36.5	1.0
Crystal-GFN	1M	51.7	51.7	51.7	$\mathbf{-1.30 \pm 2.63}$	2.09 ± 2.38	1.87 ± 0.97	0.0	0.0 (0)	0.0	0.0
Crystalformer	5M	69.9	69.4	31.8	-0.17 ± 1.48	0.70 ± 1.33	0.66 ± 1.05	1.4	0.0 (0)	28.8	3.1
DiffCSP	12M	95.7	94.8	66.2	-0.64 ± 0.96	0.28 ± 0.56	0.59 ± 0.63	2.3	0.1 (3)	29.8	8.5
DiffCSP++	12M	95.3	95.1	62.0	-0.52 ± 0.87	0.41 ± 0.44	0.69 ± 0.82	1.0	0.2 (4)	26.4	5.0
LLaMat2-CIF	7B	84.4	81.4	30.0	-0.47 ± 1.01	0.44 ± 0.71	0.54 ± 0.71	0.7	0.1 (3)	34.7	2.1
LLaMat3-CIF	8B	15.4	15.2	10.5	0.74 ± 2.69	1.71 ± 2.48	1.01 ± 1.10	0.1	0.0 (0)	2.1	0.2
SymmCD	12M	73.4	73.0	47.0	-0.02 ± 1.22	0.88 ± 1.07	0.87 ± 1.09	1.4	0.1 (2)	18.6	2.4
Ours: Without relaxation											
MP20-only ZATOM-1	80M	73.2	70.4	21.0	-0.53 ± 0.97	0.44 ± 0.64	0.42 ± 0.49	0.6	0.00 (0)	32.7	0.20 (5)
Jointly trained ZATOM-1	80M	88.5	84.4	8.1	-0.84 ± 0.89	0.19 ± 0.37	0.21 ± 0.27	2.7	0.04 (1)	51.8	0.36 (9)
Jointly trained ZATOM-1-L	160M	95.0	90.4	3.7	-0.87 ± 0.86	$\mathbf{0.09 \pm 0.21}$	0.14 ± 0.20	3.1	0.04 (1)	**71.5**	0.64 (16)
Jointly trained ZATOM-1-XL	300M	94.9	90.1	4.8	-0.91 ± 0.87	0.11 ± 0.23	0.16 ± 0.19	3.3	0.00 (0)	64.3	0.36 (9)
Jointly trained ZATOM-1-WD	80M	67.2	65.9	67.1	-0.37 ± 1.01	0.52 ± 0.67	0.47 ± 0.50	0.2	0.2 (5)	22.8	21.7 (542)
Ours: With relaxation											
Jointly trained ZATOM-1-WD	80M	88.3	86.4	**87.9**	-0.82 ± 0.84	0.11 ± 0.14	0.32 ± 0.40	4.6	**4.4 (110)**	50.6	**48.9 (1,222)**

experiments. MP20 [113] is a collection of 45,231 metastable crystal structures from the Materials Project [45], with up to 20 atoms in each material's unit cell spanning 89 different atom/element types. QM9 [111] is comprised of 130,000 stable small molecules with up to 9 heavy atoms (C, N, O, F) along with hydrogens. To ensure fair comparisons, we split the data according to prior work [113, 41]. Additionally, we report generative pretraining results for GEOM-Drugs [3], a collection of 430,000 large molecules with up to 180 atoms, following past benchmarking conventions [102, 49]. In Table 10 of Appendix F, we also employ the QMOF dataset of 14,000 metal-organic framework (MOF) structures [82] after removing structures with more than 150 atoms [49].

For predictive finetuning, we use the QM9 and Matbench datasets, the latter supporting several fundamental property prediction tasks in materials science [26]. Further, in Table 13 of Appendix F, we adopt the OMol25 (4M) [59] and MPtrj [24] datasets for molecule and material energy and force prediction, respectively. Data splits follow prior work for fair evaluation [62, 26, 59, 24]. Notably, we are also the *first* to adopt OMol25 (4M) for *larger-scale* generative pretraining experiments.

Training and hyperparameters. Our generative pretraining procedure is described in Tables 5 and 6 of Appendix D. Predictive finetuning is outlined in Table 7 of Appendix D, during which we finetune the weights dedicated to a particular task individually to minimize time to convergence.

Key inference hyperparameters include the white noise scale γ_g of each modality (following Geffner et al. [34]) and the number of integration steps T. Through an extensive hyperparameter sweep illustrated in Figure 9 of Appendix D, we find that $T = 50$ or 100 with $\gamma_g = 0.01$ generally works well for molecule and material generation, except for QM9-sized small molecules, which benefit from adding additional white noise (e.g., 50) to non-periodic atom positions to improve sample uniqueness/diversity. Like Vonessen et al. [103], $g(t) = \frac{1}{t+0.01}$ for noise scaling by default.

Evaluation metrics. Following previous work [41, 49], for generative pretraining, we evaluate ZATOM-1's ability to generate valid and realistic molecules and materials by sampling 10,000 molecules and 2,500 materials and computing validity, uniqueness, and PoseBusters sanity checks for molecules [11] and calculating validity, energy, (meta)stability, uniqueness, and novelty (i.e., (M)SUN) metrics using LeMat-GenBench [28]. These generative metrics are described in further detail in Appendix B. For predictive finetuning, we assess ZATOM-1's MAE for property, energy, and force prediction tasks, scaled/unit-converted according to literature benchmarking conventions.

Baselines. For generative pretraining, we compare ZATOM-1 trained jointly on MP20 and QM9 to material-only and molecule-only variants, as well as state-of-the-art baselines for both data sources

Table 2: **Molecule generation results on QM9.** We report (a) validity and uniqueness rates as well as (b) % pass rates on 7 sanity checks from PoseBusters for 10,000 sampled molecules. All models explicitly generate hydrogen atoms. Notably, jointly trained ZATOM-1 (80M) achieves its results using ∼400 GPU training hours compared to jointly trained ADiT (180M), which requires ∼1,200 GPU pretraining + finetuning hours.

(a) **Validity results**

Model	# Params	Validity (%) ↑	Unique (%) ↑
One-stage training			
Equivariant Diffusion	20M	91.90	98.69
Symphony	2M	83.50	97.98
Two-stage training			
GeoLDM	20M	93.80	**98.82**
QM9-only ADiT	180M	92.19	97.90
Jointly trained ADiT	180M	94.45	97.82
Ours: One-stage training			
QM9-only ZATOM-1	80M	92.88	97.71
Jointly trained ZATOM-1	80M	94.94	97.16
Jointly trained ZATOM-1-L	160M	**95.26**	96.84
Jointly trained ZATOM-1-XL	300M	95.19	97.10
Jointly trained ZATOM-1-WD	80M	94.62	97.44

(b) **PoseBusters results**

Test (% pass) ↑	Symphony	Eq. Diff.	ADiT	ZATOM-1	ZATOM-1-WD
Atoms connected	99.92	99.88	99.70	99.98	**100.00**
Bond angles	99.56	**99.98**	99.85	99.95	99.91
Bond lengths	98.72	**100.00**	99.41	99.97	99.94
Aromatic ring flat	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
Double bond flat	99.07	98.58	99.98	99.99	**100.00**
Internal energy	95.65	94.88	95.86	99.78	**99.79**
No steric clash	98.16	99.79	99.79	99.81	**99.84**

trained in one or two-stage configurations. Generated samples, as well as qualitative and quantitative evidence of ZATOM-1's cross-domain transfer learning capabilities, are shown in Appendix C.

Our experiments for materials generation using LeMat-GenBench include comparisons to: (1) five equivariant/invariant generative models adopting handcrafted multimodal product manifolds: MatterGen [119], Crystal-GFN [38], DiffCSP [46], DiffCSP++ [47], and SymmCD [60]; (2) two non-equivariant Transformer-based generative models: ADiT [49] and Crystalite [100]; and (3) four language model-based generative methods for materials: PLaID++ [114], WyFormer [52], Crystalformer [14], and LLaMat [70].

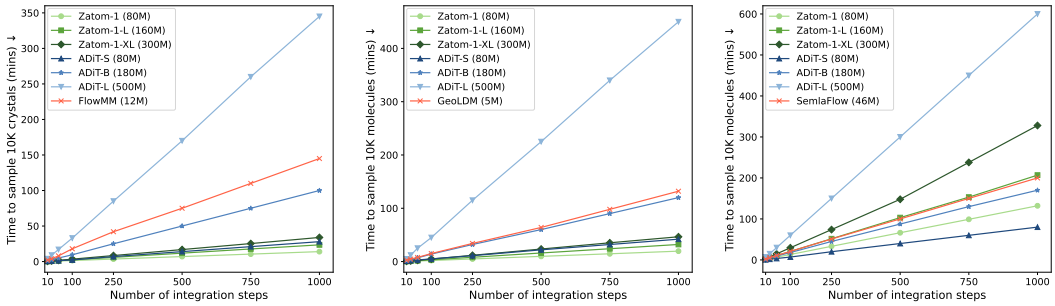
Our molecule generation experiments compare ZATOM-1 to: (1) four generative methods operating on an equivariant multimodal product manifold: Equivariant Diffusion [41], Symphony [21], EQGAT-diff [58], and SemlaFlow [43]; (2) two latent diffusion/flow matching generative models: GeoLDM [115] and ADiT [49]; and (3) non-equivariant TABASCO [103] which learns symmetries from data.

For predictive finetuning, we compare ZATOM-1 to: (1) 11 highly-optimized (i.e., hyperparameter-tuned) property prediction methods: DimeNet++ [33], EGNN [83], PaiNN [84], TorchMD-NET [96], SphereNet [66], SEGNN [9], EQGAT [57], Equiformer [62], EquiformerV2 [63], PΘNITA [6], and the Platonic Transformer [44]; (2) an unoptimized (i.e., non-hyperparameter-tuned) single-task property prediction method: AIM-STL [69]; and (3) two unoptimized *multi*-task property prediction methods: AIM-MTL-Scalar and AIM-MTL-Matrix [69]. Lastly, our energy and force prediction experiments for molecules (materials) compare ZATOM-1 to equivariant eSEN [31] (Orb-v1 [72]).

4 Results

In this section, we discuss ZATOM-1's performance in generative modeling and representation learning tasks for 3D molecules and materials, demonstrating its potential as a scalable proof-of-principle architecture for future atomistic foundation models. Additionally, we discuss potential future directions arising from this work.

State-of-the-art crystal and molecule generation. Tables 1 and 2 show that ZATOM-1 quantitatively benefits from positive transfer learning between chemical domains compared to MP20-only or QM9-only training (see Appendix C.1 for additional, data-controlled evidence). Specifically, Table 1's results demonstrate for respective (relaxed/unrelaxed) categories that ZATOM-1, namely with ZATOM-1-WD's aggressive regularization and early stopping for the relatively small MP20 dataset, generates the highest percentage of (meta)stable, unique, and novel (i.e., (M)SUN) materials that are chemically valid, advancing the state-of-the-art (relaxed) SUN rate by ∼300% with structural relaxation and the state-of-the-art (unrelaxed) MSUN rate by ∼250% without relaxation. Similarly, Table 2's QM9 results highlight that ∼99% of ZATOM-1's generated molecules pass all PoseBusters sanity checks, while ∼95% of ADiT's molecules satisfy each check, establishing a new state-of-the-art. In line with Table 2's results, Table 3 shows that ZATOM-1 is the first generative model to reach a



(a) Crystals – MP20 (b) Molecules – QM9 (c) Molecules – GEOM-Drugs

Figure 2: **ZATOM-1 is significantly faster than equivariant and latent diffusion models.** We plot the number of integration steps for ZATOM-1, equivariant diffusion FlowMM/SemlaFlow, and latent diffusion ADiT/GeoLDM vs. time to generate 10,000 samples on a single NVIDIA GPU. ZATOM-1 scales better with the number of integration steps compared to equivariant and latent diffusion models.

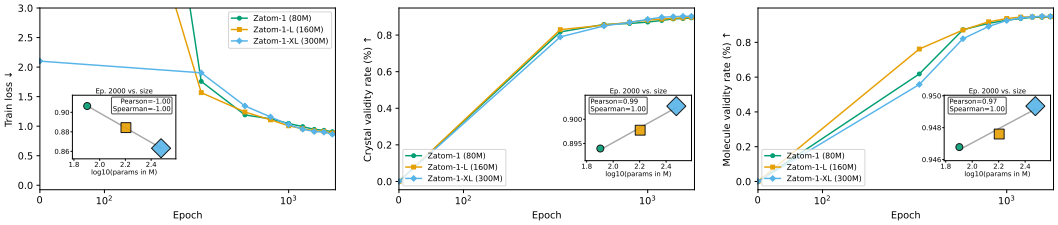


Figure 3: **Scaling up ZATOM-1 predictably improves performance.** We show the effect of increasing the number of ZATOM-1 model parameters on training loss and generation validity rates over 2,000 epochs. Correlations for training loss and validity rates at epoch 2,000 vs. ZATOM-1 parameters (in Millions) demonstrate ZATOM-1's strong scaling potential (Pearson's $r > 0.96$).

maximum-possible PoseBusters validity rate of 94% for larger, GEOM-Drugs-like molecules [103], demonstrating the strength of ZATOM-1's molecule & material pretraining for generating large, *drug-like* molecules. See Appendix E.2 for PLATOM-1's results and Table 15 of Appendix F for additional GEOM-Drugs results.

ZATOM-1 scales better than baselines. For both QM9 and MP20 sample generation, Figure 2 illustrates that ZATOM-1 offers an order of magnitude faster inference speeds than ADiT, a state-of-the-art baseline, and also shows that ZATOM-1's speeds scale better for GEOM-Drugs sampling. Likewise, Figure 3 highlights that ZATOM-1's training and validation metrics improve predictably as the model's size increases. This suggests that parallel data and model scaling (e.g., with OMol25 100M) may improve performance further, which, due to resource constraints, we defer to future work.

Chemical transfer learning improves predictions. Table 4 demonstrates that ZATOM-1 achieves state-of-the-art *multi-task* property prediction performance for QM9. For the majority of QM9's target properties, these results further mark the *first* instance in SciML of positive transfer learning for prediction tasks and zero-shot generalization via joint generative pretraining, indicating that the later trunk layers of ZATOM-1 ($K = L$ and $K = 3L/4$) excel in 3D molecular tasks, while its earlier trunk layers ($K = L/4$ and $K = L/2$) are better suited for (zero-shot) 3D material tasks. Importantly, while non-pretrained ZATOM-1 often outperforms the comparable AIM-MTL baselines, with joint generative pretraining ($K = L$), ZATOM-1 *always* strongly surpasses or matches these baselines.

Additional results. Additional results for MOF generation, *larger-scale* pretraining on OMol25 (4M), and property/energy and force prediction are provided in Appendix F on the path to developing unified generative/predictive machine learning-based interatomic potentials (MLIPs) [109]. Notably, ZATOM-1 achieves competitive performance in materials force prediction against Orb-v1 and successfully balances molecular and material property prediction tasks through additional (supervised) cross-domain finetuning. To our best knowledge, we are also the *first* to enable *non-equilibrium* 3D

Table 3: **Molecule generation results on GEOM-DRUGS.** Validity, uniqueness, and % pass rates on PoseBusters for 10,000 sampled molecules. ZATOM-1 matches or exceeds state-of-the-art equivariant/latent diffusion baselines and generates the largest number of PoseBusters-valid samples.

Metric (% pass) ↑	EQGAT-diff	SemlaFlow	ADiT	TABASCO	ZATOM-1	GEOM-Drugs
Validity	94.6	93.9	95.3	**97.6**	93.6	-
Uniqueness	**100.0**	**100.0**	**100.0**	99.03	99.93	-
Atoms connected	84.4	92.3	93.0	**99.9**	99.6	-
Bond angles	86.9	94.8	92.3	99.2	**99.6**	-
Bond lengths	87.0	94.6	92.5	99.4	**99.7**	-
Aromatic ring flat	87.0	94.9	95.4	**100.0**	**100.0**	-
Double bond flat	87.0	94.2	95.3	99.8	**100.0**	-
Internal energy	86.8	94.8	91.3	**99.4**	99.3	-
No steric clash	82.9	92.0	91.8	94.3	**96.4**	-
PoseBusters valid	59.7	87.5	85.3	91.6	**94.1**	94.0

Table 4: **Property prediction for 3D molecules (QM9) / materials (Matbench), with zero-shot multi-task material predictions.** Results are reported as *test* mean absolute errors (for ZATOM-1, with standard deviations over three runs with different random seeds). † denotes using different data partitions. / represents a task for which both (QM9) molecule and (Matbench) material property labels are available. NN denotes pretraining using a Noisy Nodes self-supervised objective [118]. The best (or second-best) results for each optimized (i.e., hyperparameter-tuned) or unoptimized (i.e., non-hyperparameter-tuned) category are in **bold** (or underlined). Notably, ZATOM-1 (80M ❄ + 20M 🔥) and AIM-MTL are the only multi-task learning (MTL) methods represented in these results, which they achieved with only ∼100 GPU hours of additional finetuning for molecule property prediction, signifying their computational efficiency and expressivity. In contrast, single-task learning (STL) baselines such as EquiformerV2 (11M) traditionally require ∼150 GPU training hours *for each task*, not including their extensive hyperparameter optimization per task. Further, for the majority of QM9's properties, jointly pretrained ZATOM-1 offers the *first* example of positive transfer learning between molecules and materials for prediction tasks driven by cross-domain generative pretraining.

Model / Metrics ↓	# Params	α (a_0^3)	$\Delta\varepsilon$ (meV)	$\varepsilon_{\text{HOMO}}$ (meV)	$\varepsilon_{\text{LUMO}}$ (meV)	μ (D)	C_ν (cal/mol K)	G^{ATOM} (meV)	H^{ATOM} (meV)	R^2 (a_0^2)	U^{ATOM} (meV)	U_0^{ATOM} (meV)	$ZPVE$ (meV)
Optimized single-task learning													
DimeNet++†	5M	.044	33.0 / **199**	25	20	.030	.023	8.00	7.00	.331	6.00	6.00	1.21
EGNN†	1M	.071	48.0 / —	29	25	.029	.031	12.0	12.0	.106	12.0	11.0	1.55
PaiNN	1M	.045	46.0 / —	28	20	.012	.024	**7.35**	**5.98**	.066	**5.83**	**5.85**	1.28
TorchMD-NET	7M	.059	36.0 / —	20 (16 w/ NN)	18 (13 w/ NN)	.011	.026	7.62	6.16	**.033**	6.38	6.15	1.84
SphereNet	2M	.046	32.0 / —	23	18	.026	**.021**	8.00	6.00	.292	7.00	6.00	**1.12**
SEGNN†	1M	.060	42.0 / —	24	21	.023	.031	15.0	16.0	.660	13.0	15.0	1.62
EQGAT	-	.053	32.0 / —	20	16	.011	.024	23.0	24.0	.382	25.0	25.0	2.00
Equiformer	4M	.046	30.0 / —	15	14	.011	.023	7.63	6.63	.251	6.74	6.59	1.26
EquiformerV2	11M	.050	**29.0** / —	**14**	**13**	.010	.023	7.57	6.22	.186	6.49	6.17	1.47
P6NITA	-	**.038**	30.4 / —	16	15	.012	.024	8.63	8.04	.235	8.67	8.31	1.29
Platonic Transformer	-	.049	37.4 / —	22	17	**.010**	.024	12.0	12.0	.222	11.9	13.0	1.30
Unoptimized single-task learning													
AIM-STL†	-	.181	— / —	61	54	**.067**	.072	66.2	63.9	**.503**	64.2	58.8	4.54
Unoptimized multi-task learning													
AIM-MTL-Scalar†	-	.268	— / —	59	71	.089	.103	113	112	4.18	112	112	9.82
AIM-MTL-Matrix†	-	.251	— / —	61	72	.088	.103	112	118	4.07	116	116	12.2
Non-pretrained ZATOM-1 (OURS)	80M ❄ + 20M 🔥	.140±.001	72.3±0.1 / 3358±6	54±0.0	49±0.0	.157±.001	.064±.000	45.5±0.2	44.7±0.2	4.72±0.02	45.2±0.2	45.8±0.2	2.89±0.01
QM9-pretrained ZATOM-1 (OURS)	80M ❄ + 20M 🔥	.095±.001	49.5±0.3 / 5297±29	36±0.2	34±0.1	.100±.000	.044±.000	23.1±0.2	23.1±0.2	3.63±0.01	23.5±0.2	22.6±0.2	2.04±0.01
Jointly pretrained ZATOM-1, $K = L/4$ (OURS)	80M ❄ + 20M 🔥	.093±.000	49.1±0.1 / **2522±11**	36±0.0	34±0.1	.111±.000	.042±.000	22.8±0.2	22.6±0.1	3.62±0.01	**22.6±0.2**	**22.1±0.2**	**1.97±0.00**
Jointly pretrained ZATOM-1, $K = L/2$ (OURS)	80M ❄ + 20M 🔥	.099±.001	51.2±0.3 / 2640±8	37±0.1	34±0.2	.121±.001	.044±.000	24.1±0.1	23.3±0.2	3.96±0.01	23.5±0.2	23.4±0.1	2.01±0.01
Jointly pretrained ZATOM-1, $K = 3L/4$ (OURS)	80M ❄ + 20M 🔥	.096±.000	49.1±0.2 / 4327±14	36±0.2	33±0.1	.125±.000	.044±.000	**22.0±0.2**	**22.3±0.2**	4.00±0.01	22.7±0.2	22.4±0.2	1.99±0.00
Jointly pretrained ZATOM-1 (OURS)	80M ❄ + 20M 🔥	**.091±.001**	**46.2±0.2** / 3891±20	**34±0.1**	**32±0.2**	.090±.001	**.041±.000**	23.3±0.4	23.0±0.4	3.29±0.01	24.0±0.4	23.3±0.4	1.98±0.01

molecule generation via the chemically diverse OMol25 dataset (n.b., containing metal complexes, biomolecules, etc.), representing a 10x data scale-up over prior GEOM-Drugs benchmarking [49].

5 Conclusion

In this work, we introduced ZATOM-1, a general-purpose model architecture for 3D molecules and materials, as a step towards a true foundation model architecture for 3D chemistry. Empirical results show that ZATOM-1 achieves state-of-the-art 3D molecule and materials generation, with fast training and inference speeds. Additionally, it demonstrates the advantages of multi-domain generative pretraining for downstream prediction tasks, providing state-of-the-art multi-task molecular property predictions for QM9, which marks SciML's first instance of positive inter-domain transfer for prediction tasks through cross-domain generative pretraining. Supplementary experiments, including those with the QMOF and OMol25 datasets, confirm that ZATOM-1 can also learn effective representations of complex atomistic systems, in spite of its current limitations (see Appendix D). Notably, given ZATOM-1's all-atom architectural scalability and versatility, future work performing parallel model and data scaling, in addition to integrating new data domains, tasks, and modalities (e.g., proteins or RNA), may unlock new foundational capabilities within and beyond the chemical sciences. As a generalized model for 3D chemistry, ZATOM-1's code and weights are freely available.

Acknowledgments and Disclosure of Funding

This research used resources of the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231 using AI4Sci@NERSC award NERSC DDR-ERCAP0036206 awarded to AM. NBE would like to acknowledge support from the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, EXPRESS: 2025 Exploratory Research for Extreme-Scale Science program, and the Scientific Discovery through Advanced Computing (SciDAC) program, under Contract Number DE-AC02-05CH11231 at Berkeley Lab. Lastly, we would also like to thank Luis Pinto and Ali Ramlaoui for their insightful feedback on the ZATOM-1 model architecture.

References

[1] J. Abramson, J. Adler, J. Dunger, R. Evans, T. Green, A. Pritzel, O. Ronneberger, L. Willmore, A. J. Ballard, J. Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024.

[2] M. S. Albergo and E. Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *The Eleventh International Conference on Learning Representations*, 2023.

[3] S. Axelrod and R. Gomez-Bombarelli. Geom, energy-annotated molecular conformations for property prediction and molecular generation. *Scientific Data*, 2022.

[4] I. Batatia, P. Benner, Y. Chiang, A. M. Elena, D. P. Kovács, J. Riebesell, X. R. Advincula, M. Asta, M. Avaylon, W. J. Baldwin, F. Berger, N. Bernstein, A. Bhowmik, F. Bigi, S. M. Blau, V. Cărare, M. Ceriotti, S. Chong, J. P. Darby, S. De, F. Della Pia, V. L. Deringer, R. Elijošius, Z. El-Machachi, E. Fako, F. Falcioni, A. C. Ferrari, J. L. A. Gardner, M. J. Gawkowski, A. Genreith-Schriever, J. George, R. E. A. Goodall, J. Grandel, C. P. Grey, P. Grigorev, S. Han, W. Handley, H. H. Heenen, K. Hermansson, C. H. Ho, S. Hofmann, C. Holm, J. Jaafar, K. S. Jakob, H. Jung, V. Kapil, A. D. Kaplan, N. Karimitari, J. R. Kermode, P. Kourtis, N. Kroupa, J. Kullgren, M. C. Kuner, D. Kuryla, G. Liepuoniute, C. Lin, J. T. Margraf, I.-B. Magdău, A. Michaelides, J. H. Moore, A. A. Naik, S. P. Niblett, S. W. Norwood, N. O'Neill, C. Ortner, K. A. Persson, K. Reuter, A. S. Rosen, L. A. M. Rosset, L. L. Schaaf, C. Schran, B. X. Shi, E. Sivonxay, T. K. Stenczel, C. Sutton, V. Svahn, T. D. Swinburne, J. Tilly, C. van der Oord, S. Vargas, E. Varga-Umbrich, T. Vegge, M. Vondrák, Y. Wang, W. C. Witt, T. Wolf, F. Zills, and G. Csányi. A foundation model for atomistic materials chemistry. *The Journal of Chemical Physics*, 163(18):184110, 2025. ISSN 0021-9606. doi: 10.1063/5.0297006.

[5] E. Bekkers, M. W. Lafarge, M. Veta, K. A. Eppenhof, J. P. Pluim, and R. Duits. Roto-translation covariant convolutional networks for medical image analysis. *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2018.

[6] E. J. Bekkers, S. Vadgama, R. Hesselink, P. A. V. der Linden, and D. W. Romero. Fast, expressive $\mathrm{SE}(n)$ equivariant networks through weight-sharing in position-orientation space. In *The Twelfth International Conference on Learning Representations*, 2024.

[7] A. Blanco-Gonzalez, A. Cabezon, A. Seco-Gonzalez, D. Conde-Torres, P. Antelo-Riveiro, A. Pineiro, and R. Garcia-Fandino. The role of ai in drug discovery: challenges, opportunities, and strategies. *Pharmaceuticals*, 16(6):891, 2023.

[8] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[9] J. Brandstetter, R. Hesselink, E. van der Pol, E. J. Bekkers, and M. Welling. Geometric and physical quantities improve e(3) equivariant message passing. In *International Conference on Learning Representations*, 2022.

[10] R. Briq, M. Kamp, O. Fried, S. Cohen, and S. Kesselheim. The amazing stability of flow matching. In *EurIPS Workshop on Principles of Generative Modeling (PriGM)*, 2025.

[11] M. Buttenschoen, G. M. Morris, and C. M. Deane. Posebusters: Ai-based docking methods fail to generate physically valid poses or generalise to novel sequences. *Chemical Science*, 15 (9):3130–3139, 2024.

[12] M. Buttenschoen, Y. Ziv, G. M. Morris, and C. Deane. An evaluation of unconditional 3d molecular generation methods. In *ICLR 2025 Workshop on Generative and Experimental Perspectives for Biomolecular Design*, 2025. URL `https://openreview.net/forum?id=ZU1m9Y4tYy`.

[13] A. Campbell, J. Yim, R. Barzilay, T. Rainforth, and T. Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. In R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 5453–5512. PMLR, 21–27 Jul 2024.

[14] Z. Cao, X. Luo, J. Lv, and L. Wang. Space group informed transformer for crystalline materials generation. *Science Bulletin*, 2025.

[15] G. Cesa, L. Lang, and M. Weiler. A program to build E(N)-equivariant steerable CNNs. *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/pdf?id=WE4qe9xlnQw.

[16] J. Chang and J. Ye. Bidirectional generation of structure and properties through a single molecular foundation model. *Nat. Commun.*, 15:2323, 2024.

[17] A. H. Cheng, C. Sun, and A. Aspuru-Guzik. Scalable autoregressive 3d molecule generation. *arXiv preprint arXiv:2505.13791*, 2025.

[18] B. Cheng. Response matching for generating materials and molecules. *Journal of Chemical Theory and Computation*, 20(20):9259–9266, 2024.

[19] J. Choi, G. Nam, J. Choi, and Y. Jung. A perspective on foundation models in chemistry. *JACS Au*, 5(4):1499–1518, 2025. doi: 10.1021/jacsau.4c01160.

[20] T. S. Cohen and M. Welling. Group equivariant convolutional networks. *International Conference on Machine Learning (ICML)*, 2016.

[21] A. Daigavane, S. E. Kim, M. Geiger, and T. Smidt. Symphony: Symmetry-equivariant point-centered spherical harmonics for 3d molecule generation. In *The Twelfth International Conference on Learning Representations*, 2024.

[22] T. Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. In *The Twelfth International Conference on Learning Representations*, 2024.

[23] D. W. Davies, K. T. Butler, A. J. Jackson, J. M. Skelton, K. Morita, and A. Walsh. Smact: Semiconducting materials by analogy and chemical theory. *Journal of Open Source Software*, 4(38):1361, 2019.

[24] B. Deng, P. Zhong, K. Jun, J. Riebesell, K. Han, C. J. Bartel, and G. Ceder. Chgnet as a pretrained universal neural network potential for charge-informed atomistic modelling. *Nature Machine Intelligence*, 5(9):1031–1041, 2023.

[25] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 2019.

[26] A. Dunn, Q. Wang, A. Ganose, D. Dopp, and A. Jain. Benchmarking materials property prediction methods: the matbench test set and automatminer reference algorithm. *npj Computational Materials*, 6(1):138, 2020.

[27] I. Dunn and D. R. Koes. Flowmol3: Flow matching for 3d de novo small-molecule generation. *arXiv preprint arXiv:2508.12629*, 2025.

[28] A. Duval, S. Betala, S. P. Gleason, A. Xu, G. Channing, D. Levy, A. Ramlaoui, C. Fourrier, C. K. Joshi, N. Kazeev, et al. Lemat-genbench: Bridging the gap between crystal generation and materials discovery. In *NeurIPS AI for Accelerated Materials Design Workshop*, 2025.

[29] S. Edamadaka, S. Yang, and R. Gomez-Bombarelli. Universally converging representations of matter across scientific foundation models. In *NeurIPS AI for Accelerated Materials Design Workshop*, 2025.

[30] N. C. Frey, R. Soklaski, S. Axelrod, S. Samsi, R. Gomez-Bombarelli, C. W. Coley, and V. Gadepally. Neural scaling of deep chemical models. *Nature Machine Intelligence*, 5(11): 1297–1305, 2023.

[31] X. Fu, B. M. Wood, L. Barroso-Luque, D. S. Levine, M. Gao, M. Dzamba, and C. L. Zitnick. Learning smooth and expressive interatomic potentials for physical property prediction. In *Forty-second International Conference on Machine Learning*, 2025.

[32] R. Gao, E. Hoogeboom, J. Heek, V. de Bortoli, K. P. Murphy, and T. Salimans. Diffusion meets flow matching: Two sides of the same coin. https://diffusionflow.github.io/, 2024.

[33] J. Gasteiger, S. Giri, J. T. Margraf, and S. Günnemann. Fast and uncertainty-aware directional message passing for non-equilibrium molecules. *arXiv preprint arXiv:2011.14115*, 2020.

[34] T. Geffner, K. Didi, Z. Zhang, D. Reidenbach, Z. Cao, J. Yim, M. Geiger, C. Dallago, E. Kucukbenli, A. Vahdat, and K. Kreis. Proteina: Scaling flow-based protein structure

generative models. In *The Thirteenth International Conference on Learning Representations*, 2025.

[35] R. W. Grosse-Kunstleve, N. K. Sauter, and P. D. Adams. Numerically stable algorithms for the computation of reduced unit cells. *Foundations of Crystallography*, 2004.

[36] C. Harris, K. Didi, A. Jamasb, C. Joshi, S. Mathis, P. Lio, and T. Blundell. Posecheck: Generative models for 3d structure-based drug design produce unrealistic poses. In *NeurIPS 2023 Generative AI and Biology (GenBio) Workshop*, 2023.

[37] T. Hayes, R. Rao, H. Akin, N. J. Sofroniew, D. Oktay, Z. Lin, R. Verkuil, V. Q. Tran, J. Deaton, M. Wiggert, et al. Simulating 500 million years of evolution with a language model. *Science*, 387(6736):850–858, 2025.

[38] A. Hernández-García, A. Volokhova, A. A. Duval, Y. Bengio, D. Sharma, P. L. Carrier, M. Koziarski, V. Schmidt, et al. Crystal-gfn: sampling materials with desirable properties and constraints. In *NeurIPS AI for Accelerated Materials Design Workshop*, 2023.

[39] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 2020.

[40] P. Höllmer, T. Egg, M. Martirossyan, E. Fuemmeler, Z. Shui, A. Gupta, P. Prakash, A. Roitberg, M. Liu, G. Karypis, M. Transtrum, R. Hennig, E. B. Tadmor, and S. Martiniani. Open materials generation with stochastic interpolants. In *Forty-second International Conference on Machine Learning*, 2025. URL `https://openreview.net/forum?id=gHGrzxFujU`.

[41] E. Hoogeboom, V. G. Satorras, C. Vignac, and M. Welling. Equivariant diffusion for molecule generation in 3d. In *International conference on machine learning*, pages 8867–8887. PMLR, 2022.

[42] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. LoRA: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=nZeVKeeFYf9`.

[43] R. Irwin, A. Tibo, J. P. Janet, and S. Olsson. Semlaflow – efficient 3d molecular generation with latent attention and equivariant flow matching. In *The 28th International Conference on Artificial Intelligence and Statistics*, 2025.

[44] M. M. Islam, R. Anand, D. R. Wessels, F. de Kruiff, T. P. Kuipers, R. Ying, C. I. Sánchez, S. Vadgama, G. Bökman, and E. J. Bekkers. Platonic transformers: A solid choice for equivariance. *arXiv preprint arXiv:2510.03511*, 2025.

[45] A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, et al. Commentary: The materials project: A materials genome approach to accelerating materials innovation. *APL materials*, 1(1), 2013.

[46] R. Jiao, W. Huang, P. Lin, J. Han, P. Chen, Y. Lu, and Y. Liu. Crystal structure prediction by joint equivariant diffusion. *Advances in Neural Information Processing Systems*, 36: 17464–17497, 2023.

[47] R. Jiao, W. Huang, Y. Liu, D. Zhao, and Y. Liu. Space group constrained crystal generation. In *The Twelfth International Conference on Learning Representations*, 2024.

[48] X. Jin, K. M. Jablonka, E. Moubarak, Y. Li, and B. Smit. Mofchecker: a package for validating and correcting metal–organic framework (mof) structures. *Digital Discovery*, 2025.

[49] C. K. Joshi, X. Fu, Y.-L. Liao, V. Gharakhanyan, B. K. Miller, A. Sriram, and Z. W. Ulissi. All-atom diffusion transformers: Unified generative modelling of molecules and materials. In *Forty-second International Conference on Machine Learning*, 2025.

[50] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.

[51] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[52] N. Kazeev, W. Nong, I. Romanov, R. Zhu, A. E. Ustyuzhanin, S. Yamazaki, and K. Hippalgaonkar. Wyckoff transformer: Generation of symmetric crystals. In *Forty-second International Conference on Machine Learning*, 2025.

[53] S. Kitagawa et al. Metal–organic frameworks (mofs). *Chemical Society Reviews*, 43(16): 5415–5418, 2014.

[54] T. Kreiman, Y. Bai, F. Atieh, E. Weaver, E. Qu, and A. S. Krishnapriyan. Transformers discover molecular structure without graph priors. *arXiv preprint arXiv:2510.02259*, 2025.

[55] A. Krishnapriyan, A. Gholami, S. Zhe, R. Kirby, and M. W. Mahoney. Characterizing possible failure modes in physics-informed neural networks. *Advances in neural information processing systems*, 34:26548–26560, 2021.

[56] G. Landrum. Rdkit documentation. *Release*, 1(1-79):4, 2013.

[57] T. Le, F. Noe, and D.-A. Clevert. Representation learning on biomolecular structures using equivariant graph attention. In *The First Learning on Graphs Conference*, 2022.

[58] T. Le, J. Cremer, F. Noe, D.-A. Clevert, and K. T. Schütt. Navigating the design space of equivariant diffusion-based generative models for de novo 3d molecule generation. In *The Twelfth International Conference on Learning Representations*, 2024.

[59] D. S. Levine, M. Shuaibi, E. W. C. Spotte-Smith, M. G. Taylor, M. R. Hasyim, K. Michel, I. Batatia, G. Csányi, M. Dzamba, P. Eastman, et al. The open molecules 2025 (omol25) dataset, evaluations, and models. *arXiv preprint arXiv:2505.08762*, 2025.

[60] D. Levy, S. S. Panigrahi, S.-O. Kaba, Q. Zhu, K. L. K. Lee, M. Galkin, S. Miret, and S. Ravanbakhsh. SymmCD: Symmetry-preserving crystal generation with diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025.

[61] A. C. Li, M. Prabhudesai, S. Duggal, E. Brown, and D. Pathak. Your diffusion model is secretly a zero-shot classifier. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2206–2217, October 2023.

[62] Y.-L. Liao and T. Smidt. Equiformer: Equivariant graph attention transformer for 3d atomistic graphs. In *The Eleventh International Conference on Learning Representations*, 2023.

[63] Y.-L. Liao, B. M. Wood, A. Das, and T. Smidt. Equiformerv2: Improved equivariant transformer for scaling to higher-degree representations. In *The Twelfth International Conference on Learning Representations*, 2024.

[64] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023.

[65] S. Liu, H. Wang, W. Liu, J. Lasenby, H. Guo, and J. Tang. Pre-training molecular graph representation with 3d geometry. In *International Conference on Learning Representations*, 2022.

[66] Y. Liu, L. Wang, M. Liu, Y. Lin, X. Zhang, B. Oztekin, and S. Ji. Spherical message passing for 3d molecular graphs. In *International Conference on Learning Representations*, 2022.

[67] S. Lu, H. Lin, L. Yao, Z. Gao, X. Ji, Y. Liang, W. E, L. Zhang, and G. Ke. Unified cross-scale 3d generation and understanding via autoregressive modeling. In *ICLR 2026 Workshop on Foundation Models for Science: Real-World Impact and Science-First Design*, 2026.

[68] B. K. Miller, R. T. Q. Chen, A. Sriram, and B. M. Wood. FlowMM: Generating materials with riemannian flow matching. In *Forty-first International Conference on Machine Learning*, 2024.

[69] M. Minot and G. Schneider. AIM: Adaptive intervention for deep multi-task learning of molecular properties. In *NeurIPS 2025 AI for Science Workshop*, 2025.

[70] V. Mishra, S. Singh, D. Ahlawat, M. Zaki, V. Bihani, H. S. Grover, B. Mishra, S. Miret, N. Krishnan, et al. Foundational large language models for materials research. *arXiv preprint arXiv:2412.09560*, 2024.

[71] A. Morehead, L. Atanackovic, A. Hegde, Y. Wang, F. Boadu, J. Selvaraj, A. Tong, A. Krishnapriyan, and J. Cheng. Flow matching for generative modeling in bioinformatics and computational biology. *Nature Machine Intelligence*, 2026.

[72] M. Neumann, J. Gin, B. Rhodes, S. Bennett, Z. Li, H. Choubisa, A. Hussey, and J. Godwin. Orb: A fast, scalable neural network potential. *arXiv preprint arXiv:2410.22570*, 2024.

[73] J. Nguyen, M. Havasi, T. Berrada, L. Zettlemoyer, and R. T. Chen. Oneflow: Concurrent mixed-modal and interleaved generation with edit flows. *arXiv preprint arXiv:2510.03506*, 2025.

[74] S. Nie, F. Zhu, Z. You, X. Zhang, J. Ou, J. Hu, J. Zhou, Y. Lin, J.-R. Wen, and C. Li. Large language diffusion models. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems*, 2025.

[75] A. Okhotin, M. Nakhodnov, N. Kazeev, A. E. Ustyuzhanin, and D. Vetrov. Miad: Mirage atom diffusion for de novo crystal generation. *arXiv preprint arXiv:2511.14426*, 2025.

[76] S. P. Ong, W. D. Richards, A. Jain, G. Hautier, M. Kocher, S. Cholia, D. Gunter, V. L. Chevrier, K. A. Persson, and G. Ceder. Python materials genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science*, 68:314–319, 2013.

[77] H. Park and A. Walsh. Guiding generative models to uncover diverse and novel crystals via reinforcement learning. *arXiv preprint arXiv:2511.07158*, 2025.

[78] T. Perez and R. Gomez-Bombarelli. Self-conditioned denoising for atomistic representation learning. *arXiv preprint arXiv:2603.17196*, 2026.

[79] E. O. Pyzer-Knapp, M. Manica, P. Staar, L. Morin, P. Ruch, T. Laino, J. R. Smith, and A. Curioni. Foundation models for materials discovery – current state and future directions. *npj Comput. Mater.*, 11:61, 2025. doi: 10.1038/s41524-025-01538-0.

[80] E. Qu and A. S. Krishnapriyan. The importance of being scalable: Improving the speed and accuracy of neural network interatomic potentials across chemical domains. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 139030–139053. Curran Associates, Inc., 2024. doi: 10.52202/079017-4412.

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

[82] A. S. Rosen, S. M. Iyer, D. Ray, Z. Yao, A. Aspuru-Guzik, L. Gagliardi, J. M. Notestein, and R. Q. Snurr. Machine learning the quantum-chemical properties of metal–organic frameworks for accelerated materials discovery. *Matter*, 4(5):1578–1597, 2021.

[83] V. G. Satorras, E. Hoogeboom, and M. Welling. E(n) equivariant graph neural networks. In *International conference on machine learning*, pages 9323–9332. PMLR, 2021.

[84] K. Schütt, O. Unke, and M. Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *International conference on machine learning*, pages 9377–9388. PMLR, 2021.

[85] K. Seong, S. Ahn, S. Han, and C. Park. Multimodal crystal flow: Any-to-any modality generation for unified crystal modeling. *arXiv preprint arXiv:2602.20210*, 2026.

[86] N. Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

[87] M. Siron, I. Djafar, A. Ramlaoui, E. d. Fayette, A. Rossello, E. Fako, M. McDermott, F. Therrien, L. Barroso-Luque, F. Cipcigan, et al. Lemat-bulk: aggregating, and de-duplicating quantum chemistry materials databases. *arXiv preprint arXiv:2511.05178*, 2025.

[88] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. pmlr, 2015.

[89] Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[90] Y. Song, J. Gong, M. Xu, Z. Cao, Y. Lan, S. Ermon, H. Zhou, and W.-Y. Ma. Equivariant flow matching with hybrid probability transport for 3d molecule generation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[91] A. Sriram, B. K. Miller, R. T. Chen, and B. M. Wood. Flowllm: Flow matching for material generation with large language models as base distributions. *Advances in Neural Information Processing Systems*, 37:46025–46046, 2024.

[92] H. Stark, B. Jing, R. Barzilay, and T. Jaakkola. Harmonic self-conditioned flow matching for joint multi-ligand docking and binding site design. In *Forty-first International Conference on Machine Learning*, 2024.

[93] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[94] S. Subramanian, P. Harrington, K. Keutzer, W. Bhimji, D. Morozov, M. W. Mahoney, and A. Gholami. Towards foundation models for scientific machine learning: Characterizing scaling and transfer behavior. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 71242–71262. Curran Associates, Inc., 2023.

[95] S. Subramanian, A. Kiefer, A. Nigmetov, A. Gholami, D. Morozov, and M. W. Mahoney. On neural scaling laws for weather emulation through continual training. In *ICLR Workshop on Foundation Models for Science: Real-World Impact and Science-First Design*, 2026.

[96] P. Thölke and G. D. Fabritiis. Equivariant transformers for neural network based molecular potentials. In *International Conference on Learning Representations*, 2022.

[97] A. Tong, K. Fatras, N. Malkin, G. Huguet, Y. Zhang, J. Rector-Brooks, G. Wolf, and Y. Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. Expert Certification.

[98] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[99] P. Vélez, L. F. Polanía, Y. Yang, C. Zhang, R. Kabra, A. Arnab, and M. S. M. Sajjadi. From image to video: An empirical study of diffusion representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16948–16958, October 2025.

[100] T. H. Veljković, J. Rosenthal, I. Lončarić, and J.-W. van de Meent. Crystalite: A lightweight transformer for efficient crystal modeling. *arXiv preprint arXiv:2604.02270*, 2026.

[101] V. Venkatasubramanian. The promise of artificial intelligence in chemical engineering: Is it here, finally? *AIChE Journal*, 65(1), 2019.

[102] C. Vignac, N. Osman, L. Toni, and P. Frossard. Midi: Mixed graph and 3d denoising diffusion for molecule generation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 560–576. Springer, 2023.

[103] C. Vonessen, C. Harris, M. Cretu, and P. Lio. TABASCO: A fast, simplified model for molecular generation with improved physical quality. *Transactions on Machine Learning Research*, 2026.

[104] A. Wadell, A. Bhutani, V. Azumah, A. R. Ellis-Mohr, C. Kelly, H. Zhao, A. K. Nayak, K. Hegazy, A. Brace, H. Lin, et al. Foundation models for discovery and exploration in chemical space. *arXiv preprint arXiv:2510.18900*, 2025.

[105] H. Wang, T. Fu, Y. Du, W. Gao, K. Huang, Z. Liu, P. Chandak, S. Liu, P. Van Katwyk, A. Deac, et al. Scientific discovery in the age of artificial intelligence. *Nature*, 620(7972):47–60, 2023.

[106] M. Weiler and G. Cesa. General E(2)-equivariant steerable CNNs. *Conference on Neural Information Processing Systems (NeurIPS)*, 2019. URL `https://arxiv.org/abs/1911.08251`.

[107] M. Weiler, F. A. Hamprecht, and M. Storath. Learning steerable filters for rotation equivariant CNNs. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. URL `https://arxiv.org/abs/1711.07289`.

[108] M. Weiler, P. Forré, E. Verlinde, and M. Welling. *Equivariant and Coordinate Independent Convolutional Networks*. WORLD SCIENTIFIC, 2025. doi: 10.1142/14143. URL `https://maurice-weiler.gitlab.io/#cnn_book`.

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

[110] M. Wortsman, P. J. Liu, L. Xiao, K. E. Everett, A. A. Alemi, B. Adlam, J. D. Co-Reyes, I. Gur, A. Kumar, R. Novak, J. Pennington, J. Sohl-Dickstein, K. Xu, J. Lee, J. Gilmer, and S. Kornblith. Small-scale proxies for large-scale transformer training instabilities. In *The Twelfth International Conference on Learning Representations*, 2024.

[111] Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, and V. Pande. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 2018.

[112] S. M. Xie, H. Pham, X. Dong, N. Du, H. Liu, Y. Lu, P. S. Liang, Q. V. Le, T. Ma, and A. W. Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. *Advances in Neural Information Processing Systems*, 36:69798–69818, 2023.

[113] T. Xie, X. Fu, O.-E. Ganea, R. Barzilay, and T. S. Jaakkola. Crystal diffusion variational autoencoder for periodic material generation. In *International Conference on Learning Representations*, 2022.

[114] A. Xu, R. Desai, L. Wang, G. Hope, and E. Ritz. Plaid++: A preference aligned language model for targeted inorganic materials design. *arXiv preprint arXiv:2509.07150*, 2025.

[115] M. Xu, A. S. Powers, R. O. Dror, S. Ermon, and J. Leskovec. Geometric latent diffusion models for 3d molecule generation. In *International Conference on Machine Learning*, pages 38592–38610. PMLR, 2023.

[116] S. Yang, K. Cho, A. Merchant, P. Abbeel, D. Schuurmans, I. Mordatch, and E. D. Cubuk. Scalable diffusion for materials generation. In *The Twelfth International Conference on Learning Representations*, 2024.

[117] X. Yi, G. Xu, Z. Zhang, L. Liu, Y. Bian, X. Xiao, and P. Zhao. Crystaldit: Simple diffusion transformers for crystal generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1462–1470, 2026.

[118] S. Zaidi, M. Schaarschmidt, J. Martens, H. Kim, Y. W. Teh, A. Sanchez-Gonzalez, P. Battaglia, R. Pascanu, and J. Godwin. Pre-training via denoising for molecular property prediction. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=tYIMtogyee`.

[119] C. Zeni, R. Pinsler, D. Zügner, A. Fowler, M. Horton, X. Fu, Z. Wang, A. Shysheya, J. Crabbé, S. Ueda, et al. A generative model for inorganic materials design. *Nature*, 639(8055):624–632, 2025.

[120] C. Zhang, H. Zhong, K. Zhang, C. Chai, R. Wang, X. Zhuang, T. Bai, Q. Jiantao, L. Cao, J. Fan, Y. Yuan, G. Wang, and C. He. Harnessing diversity for important data selection in pretraining large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=bMC1t7eLRc`.

[121] G. Zhang, Y. Li, R. Luo, P. Hu, Y. Yang, Z. Zhao, L. Li, G. Liu, Z. Wang, R. Bi, et al. Unigenx: a unified generative foundation model that couples sequence, structure and function to accelerate scientific design across proteins, molecules and materials. *arXiv preprint arXiv:2503.06687*, 2025.

[122] S. Zhang, Y. Zhao, L. Geng, A. Cohan, L. A. Tuan, and C. Zhao. Diffusion vs. autoregressive language models: A text embedding perspective. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 4273–4303, 2025.

[123] M. Zhdanov, D. Ruhe, M. Weiler, A. Lucic, J. Brandstetter, and P. Forré. Clifford-Steerable Convolutional Neural Networks. In *International Conference on Machine Learning (ICML)*, 2024. URL `https://arxiv.org/abs/2402.14730`.

[124] G. Zhou, Z. Gao, Q. Ding, H. Zheng, H. Xu, Z. Wei, L. Zhang, and G. Ke. Uni-mol: A universal 3d molecular representation learning framework. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=6K2RM6wVqKu`.

ZATOM-1 Appendices

A Related Work

A.1 Scientific foundation models

The concept of scientific foundation models, which are large, self-supervised networks pretrained on broad scientific data, has recently gained momentum in chemistry and materials science [79, 19, 29]. Recent reviews emphasize the value of training on diverse chemical datasets and using generative or multi-task objectives to build broadly useful chemical representations [79, 19, 104]. However, most existing works remain domain-specific: "foundation" efforts in molecules often focus on SMILES or 2D graph pretraining, while material science approaches largely target property prediction or synthesis planning rather than joint 3D modeling.

A.2 Generative modeling of 3D molecules

Several recent works propose equivariant diffusion or flow models for de novo 3D molecule design. Hoogeboom et al. [41] introduce an E(3)-equivariant diffusion model (EDM) that denoises continuous atom coordinates and types together. Autoregressive graph models like Symphony [21] build molecules fragment-by-fragment with spherical harmonics features. Le et al. [58] explore E(3)-equivariant diffusion losses (EQGAT-diff) to improve validity on QM9 and GEOM-Drugs. Latent diffusion approaches like GeoLDM [115] encode 3D structures into continuous latent spaces. More recent work explores the importance of equivariance: for instance, SemlaFlow uses flow matching with an equivariant graph neural network backbone [43]; FlowMol3 employs hybrid continuous-discrete flow matching with chirality-aware message passing [27]; and TABASCO shows that a non-equivariant transformer can match the validity of equivariant models while generating 3D drug-like molecules [103]. In summary, these models each focus on molecule generation but do not generalize to materials.

A.3 Generative modeling of 3D materials

Likewise, generative models for periodic crystals have been developed separately. Score-based models (diffusion) and flow matching have been applied to crystal generation: e.g., Xie et al. [113] propose CDVAE, a score-based model for generating periodic unit cells, and Jiao et al. [46] introduce DiffCSP, an E(3)-equivariant diffusion model for predicting stable crystal structures. Recent work often uses variants of normalizing flows: Miller et al. [68]'s FlowMM applies Riemannian flow matching over atom types and lattice geometry, and Yang et al. [116] show that standard (non-equivariant) diffusion on a learnable crystal representation (UniMat) can scale to large collections of structures. Another parallel thread is generative models for inorganic materials like those of Zeni et al. [119], Höllmer et al. [40], Park and Walsh [77], Okhotin et al. [75], Yi et al. [117], Seong et al. [85], and Veljković et al. [100]. FlowLLM combines a fine-tuned language model base with flow matching for crystals [91], while the All-Atom Diffusion Transformer (ADiT) unifies molecule and material generation by learning a shared latent autoencoding of both domains and diffusing in that latent space [49]. Notably, none of these approaches simultaneously addresses both molecules and materials in one model without resorting to complex two-stage training approaches, such as latent diffusion or LLM-based data generation, nor do they leverage generative pretraining for downstream tasks.

A.4 Multimodal flow/diffusion architectures

Our work joins a growing class of multimodal generative models that handle discrete atom types and continuous 3D geometries jointly. For example, Miller et al. [68] explicitly matches flows over categorical atom labels, continuous coordinates, and lattice parameters in a single framework. ADiT [49] analogously embeds all-atom molecular and crystal representations into a shared latent space for latent diffusion. These approaches stand in contrast to earlier graph or point-based generative models (e.g., EDM [41], EQGAT-diff [58]) that typically model only a small number of data modalities simultaneously. By jointly training on atom types, positions, and unit cell parameters, our approach builds on the multimodal flow concept of FlowMM while using a standard Transformer backbone for expressiveness, scalability, and speed.

A.5 Representation learning and pretraining in chemistry

Beyond full generative models, chemical representation learning has employed self-supervised pretraining on molecular graphs. For instance, GraphMVP [65] uses contrastive learning between 2D molecular topology and 3D geometry. Masked autoencoder approaches and property prediction pretraining have also been explored. However, most existing pretraining methods do not support generative modeling for sample generation [124, 118]. Even though image diffusion models have been shown to yield useful feature embeddings [61, 99], few works have investigated cross-domain generative pretraining as a widely applicable paradigm in 3D chemistry or AI for Science more broadly. Accordingly, our work frames conditional multimodal flow matching (over atom types and geometry) as a self-supervised "pretraining" task for 3D chemical systems. By pretraining on joint molecule & material data, we obtain unified embeddings transferable to property, energy, and force prediction. This idea is similar to Chang and Ye [16]'s structure-property molecular model, which learns a bidirectional generative model of molecular graphs and properties, but here we apply it broadly across both molecules and crystals.

A.6 Overview

Overall, prior work in molecular/material generation has focused either on discrete (graph/SMILES) models or on separate generative flows/diffusions for molecules or crystals [41, 113]. Our approach differs by unifying generative and predictive modeling for all-atom 3D chemistry. We draw on advances in flow matching and multimodal architecture design [90, 68, 103], and we treat generative modeling itself as a pretraining strategy. This differs from previous scientific foundation modeling efforts that focus on language or 2D graph representations [79, 19, 104]; that do not support downstream prediction tasks with learned representations [121]; that are not trained across multiple data domains *end-to-end* [67]; or that autoregressively assume a canonical token ordering for unordered sets of atoms [121, 67]. By explicitly modeling the ambient 3D space of atoms and elements together with a standardized Transformer architecture, we extend the flow-based generative paradigm to create and evaluate a single, general-purpose model architecture across 3D molecular and material domains.

B Evaluation Metrics

B.1 Model development metrics for assessing crystal generation quality

To evaluate the quality of generated crystals during (iterative) model *development*, we adopt the methodology proposed by Xie et al. [113], Miller et al. [68], and Joshi et al. [49]. This process involves generating a sample of 1,000 crystals and then assessing them based on their validity and uniqueness. These metrics are described below:

- **Structural Validity**: This metric represents the percentage of crystals where all pairwise atomic distances are at least 0.5 and the crystal volume is 0.1 or greater.
- **Compositional Validity**: In accordance with the SMACT framework [23], this is the percentage of crystal compositions that successfully maintain both charge neutrality and electronegativity balance.
- **Overall Validity**: This is the percentage of crystals that satisfy both the structural and the compositional validity criteria mentioned above. Note that this metric is adopted in Figures 3 and 9.
- **Unique**: This measures the percentage of crystals that are structurally unique. Uniqueness is determined via an all-to-all comparison using the Structure Matcher tool within PyMatGen [76].

B.2 Model evaluation metrics for assessing crystal generation quality

To assess the quality of generated crystals during (extensive) model *evaluation*, we adopt the methodology proposed by Duval et al. [28]. This process involves generating a sample of 2,500 crystals and then evaluating them based on their validity, uniqueness, novelty, energy, and stability. These metrics are detailed below:

- **Charge Validity**: This metric ensures structures are charge-balanced using oxidation state analysis and bond valence calculations.
- **Distance Validity**: This validates that atomic distances exceed minimum thresholds based on atomic radii.
- **Coordination Validity**: This checks if coordination numbers match expected values for each element.
- **Physical Validity**: This validates density, lattice parameters, crystallographic format, and symmetry.
- **Overall Validity**: This represents the percentage of crystals meeting each of the evaluation validity criteria above.
- **Unique**: This measures the percentage of *valid* crystals that are also structurally unique. Uniqueness is determined via an all-to-all comparison using PyMatGen's Structure Matcher tool.
- **Novel**: This assesses the percentage of *valid*, *unique* crystals that are also structurally novel. Novelty is determined via an all-to-all comparison between the generated crystals and the LeMat-Bulk reference set [87] using the Structure Matcher tool of PyMatGen.
- **Formation Energy**: This denotes the average formation energy across multiple MLIPs, namely Orb-v3 [81], MACE-MP [4], and UMA [109].
- **Mean Energy Above Hull**: This represents the average energy above hull across multiple MLIPs (i.e., Orb-v3, MACE-MP, UMA).
- **Relaxation Stability**: This signifies the root mean square deviation (RMSD) between original and relaxed atomic positions.
- **Stability Ratio**: This denotes the fraction of structures with energy above hull ≤ 0 eV/atom (thermodynamically stable).
- **Stable, Unique, & Novel (SUN)**: This represents the fraction of structures that are simultaneously valid, stable (energy above hull ≤ 0 eV/atom), unique, and novel, with the number of structures that meet each of these criteria shown in proceeding parentheses.
- **Metastability Ratio**: This signifies the fraction of structures with $0 <$ energy above hull $\leq$ 0.1 eV/atom (metastable).
- **Metastable, Unique, & Novel (MSUN)**: This denotes the fraction of structures that are simultaneously valid, metastable ($0 <$ energy above hull ≤ 0.1 eV/atom), unique, and novel, with the number of structures that meet each of these criteria shown in proceeding parentheses.

B.3 Model development/evaluation metrics for assessing molecule generation quality

For the evaluation of generated molecules, we utilize the protocol developed by Hoogeboom et al. [41], Daigavane et al. [21], and Joshi et al. [49]. From a sample of 10,000 molecules, we calculate rates for validity and uniqueness, and also determine success rates for seven distinct sanity checks provided by the PoseBusters toolkit [11]:

- **Validity**: This measures the percentage of generated molecules for which a canonical SMILES representation can be successfully identified using RDKit [56].
- **Uniqueness**: Among the molecules deemed valid, this metric calculates the percentage that possess a unique SMILES string.
- **All-atoms Connected**: This check quantifies the percentage of molecules in which every atom is part of a single connected component, meaning a continuous path of bonds links all atoms.
- **Reasonable Bond Angles/Lengths**: This is the fraction of molecules where all bond angles and lengths fall within a tolerance range, specifically between 0.75 and 1.25 times the standard values derived from distance geometry.
- **Aromatic Rings Flatness**: This metric counts the percentage of molecules where all atoms in 5- or 6-membered aromatic rings are positioned within 0.25Å of their nearest shared plane.

- **Double Bonds Flatness**: This is the percentage of molecules where the atoms of aliphatic carbon-carbon double bonds, along with their four neighboring atoms, lie within 0.25Å of the closest shared plane.
- **Reasonable Internal Energy**: This check determines the percentage of molecules whose calculated internal energy does not exceed 100 times the average energy observed in an ensemble of 50 conformations of the input molecule [36].
- **No Internal Steric Clash**: This metric represents the percentage of molecules where the distance between any pair of non-covalently bound atoms is greater than 0.8 times the lower bound established by distance geometry.

In summary, the validity and uniqueness metrics primarily confirm the chemical processability of the generated molecules by RDKit. In contrast, the PoseBusters sanity checks scrutinize the physical realism of the generated 3D structures, applying a range of criteria from geometric constraints to energetic plausibility.

B.4 Model evaluation metrics for assessing cross-domain transfer learning

To assess whether a model is positively leveraging cross-domain transfer learning between molecules and materials during pretraining, we extend the embedding visualization protocol developed by Joshi et al. [49]. From a random sample of 100 molecules and materials, each generated by ZATOM-1, as well as 100 molecules and materials drawn from QM9 and MP20, respectively, we calculate the Clustering & Transfer Learning Score (higher is better) as follows:

- **Clustering & Transfer Learning Score (↑)**: A summary score for 2D (e.g., PCA-derived) model embeddings. It is defined as Inter-Element Separation / (Intra-Element Spread + Dataset-Generated Shift + Crystal-Molecule Shift), so it is larger when atom types are well separated, while the same element remains aligned across source (e.g., dataset, generated) and system (e.g., molecule, material) categories.
- **Inter-Element Separation (↑)**: This is the mean pairwise Euclidean distance between the 2D centroids of element clusters (C, N, O, F). Larger values indicate stronger separation of different atom types.
- **Intra-Element Spread (↓)**: This is the mean Euclidean distance from each point to the centroid of its own element cluster. Smaller values indicate that atoms of the same type form tighter clusters.
- **Dataset-Generated Shift (↓)**: This is the mean Euclidean distance between centroids of dataset-derived and generated samples for the same element and system type. Smaller values indicate that generated atoms are embedded similarly to dataset atoms within molecules and within crystals, reducing distribution shifts between generated and real data.
- **Crystal-Molecule Shift (↓)**: This is the mean Euclidean distance between crystal and molecule centroids for the same element and source type. Smaller values indicate that, e.g., oxygen atoms in molecules and crystals occupy similar regions of the embedding space, suggesting the model has learned to leverage positive transfer learning between molecules and materials.

C Visualization

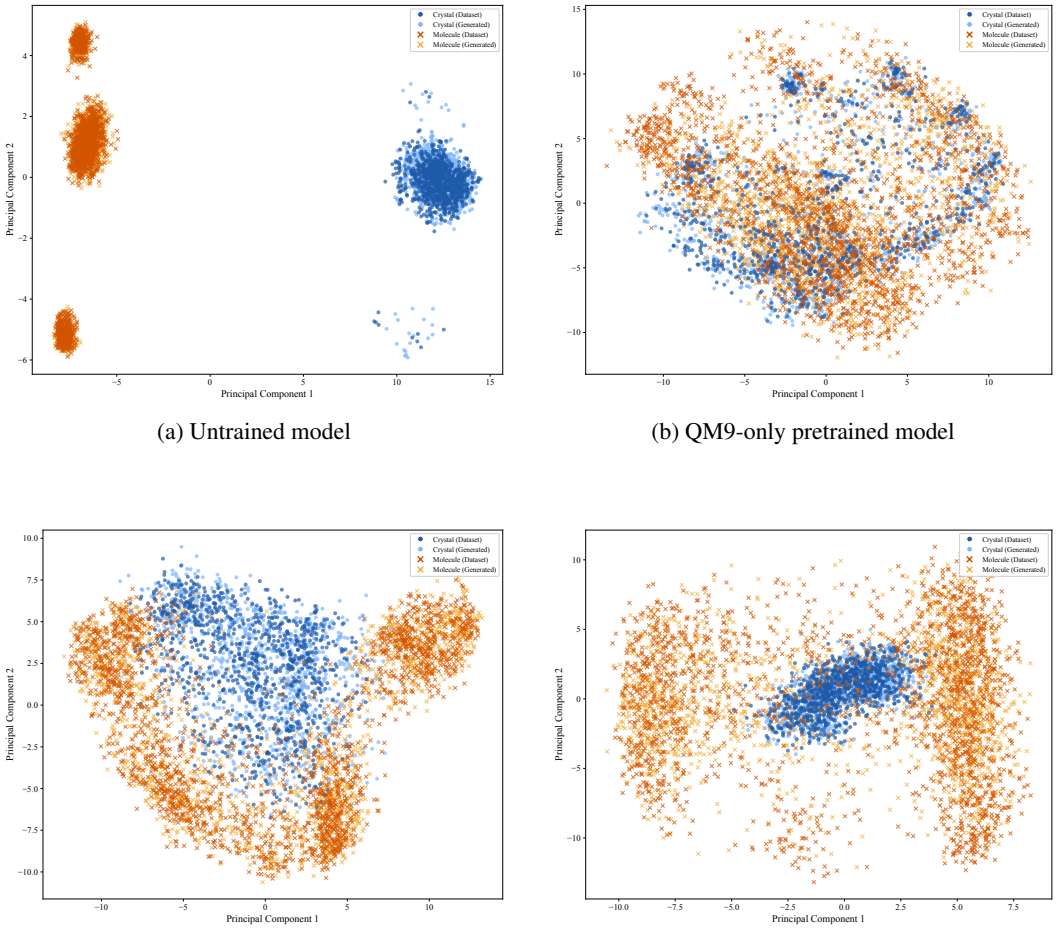
C.1 Characterizing chemical transfer learning in shared latent space

In Figure 4, following [49], we plot the first two PCA principal components of ZATOM-1's final-layer ($K = L$) Transformer trunk embeddings for 100 random samples each drawn from the MP20 and QM9 validation datasets, as well as 100 generated crystals and 100 generated molecules sampled from jointly pretrained ZATOM-1. Overall, we observe that a jointly pretrained ZATOM-1 Transformer trunk, compared to one left untrained or pretrained using only QM9 or MP20, produces a latent space with well-structured (yet overlapping) clusters between molecule and crystal atoms (including hydrogens for molecules), suggesting positive representation transfer has occurred between these two chemical domains.

To quantify and further characterize this observation, in Figure 5, we plot the same PCA of ZATOM-1's trunk embeddings but now only retain carbon, nitrogen, oxygen, and fluorine atoms, as these atoms appear in both QM9 molecules and MP20 crystals, allowing us to analyze how their representations compare across non-periodic and periodic systems. For this experiment, we propose a new metric called the **Clustering & Transfer Learning Score** (defined in Appendix B.4), which assesses how well a model's embeddings produce well-separated clusters of different elements while ensuring the same element remains aligned across source (e.g., dataset, generated) and system (e.g., molecule, material) categories. Accordingly, Figure 5 demonstrates clearly the following two patterns. (**1**) Embeddings taken from a trunk pretrained on either QM9 or MP20 form *less* well-structured molecule and material clusters, with a quantitatively higher intra-element spread and crystal-molecule shift, while a jointly pretrained trunk produces embeddings that achieve the *best* Clustering & Transfer Learning Score (n.b., which remains the *same best score* of 0.51 when jointly pretraining on an equal amount of molecule + material data (13.5k + 13.5k) as the MP20-only or QM9-only baseline models, suggesting that the *cross-domain diversity* of ZATOM-1's pretraining data [112, 120] is the driving factor behind its strong clustering and transfer learning results). (**2**) For a jointly pretrained model, principal component 1 largely distinguishes between crystals (clustered between -2 and 4) and molecules, while principal component 2, generally speaking, relates to an atom's type. Most notably, for a jointly pretrained trunk, oxygen (and, to some extent, fluorine) atoms show similar latent representations whether they appear in molecules or crystals, suggesting that ZATOM-1's latent space captures fundamental chemical properties that transfer across both chemical domains. This shared representation of oxygen, a key atom in both datasets, may partially explain ZATOM-1's successful joint pretraining with periodic and non-periodic systems.

C.2 Generated crystals, molecules, and MOFs

Figures 6 and 7 show representative crystals and small molecules, respectively, sampled from ZATOM-1 pretrained jointly on the QM9 and MP20 datasets. Additionally, Figure 8 visualizes MOFs generated by ZATOM-1 pretrained on the QMOF dataset. Overall, ZATOM-1 generates compositionally diverse crystals across multiple spacegroups and chemically valid, structurally diverse molecules, indicating that it learns a rich generative latent space rather than a redundant set of representations.



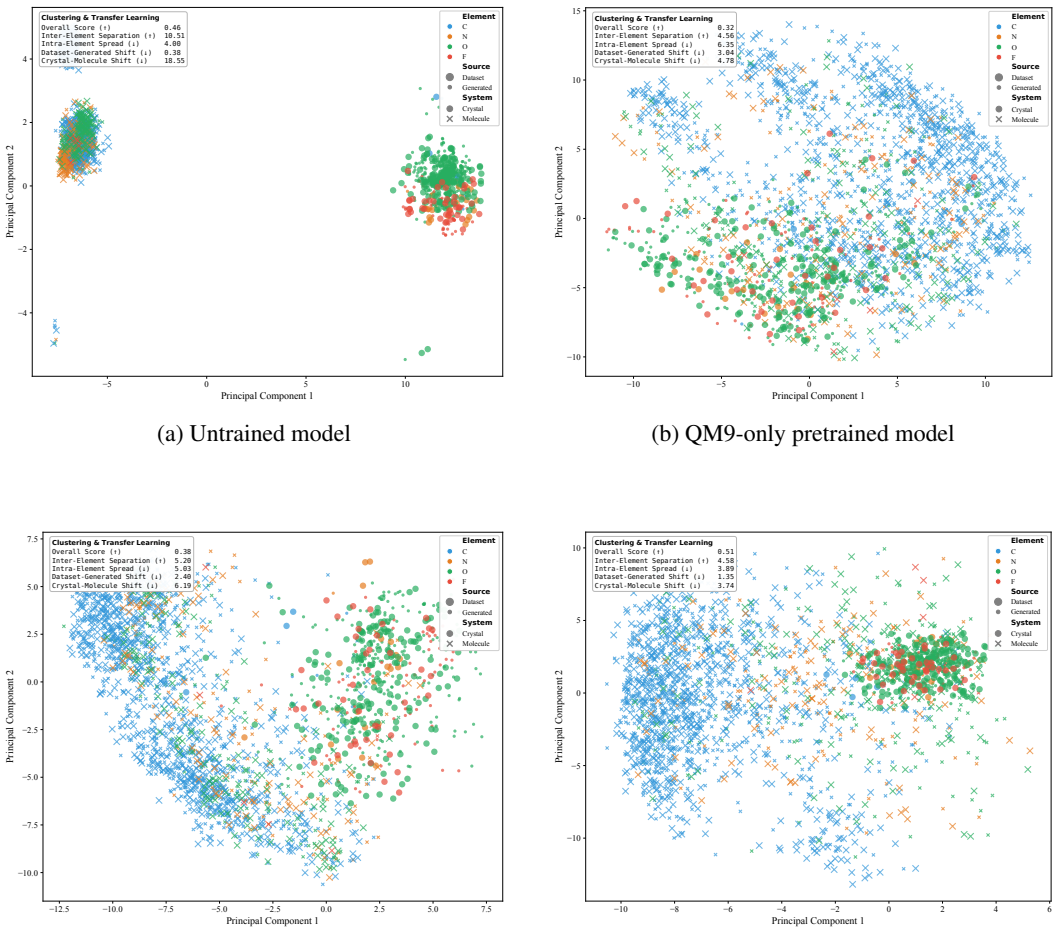
(a) Untrained model

(b) QM9-only pretrained model

(c) MP20-only pretrained model

(d) Jointly pretrained model

Figure 4: PCA plots of latent embeddings from ZATOM-1's Transformer trunk for 100 data points from the MP20 and QM9 datasets, as well as 100 molecules/materials each generated by ZATOM-1. Each point denotes an atom (including hydrogens for molecules), with its shape representing its chemical system type and its primary color indicating whether it comes from real or generated data. **(a) Embeddings taken from an untrained trunk are distributed in an overly clustered and chemically-segmented manner. (b-c) Embeddings taken from a trunk pretrained on either QM9 or MP20 form less well-structured molecule and material clusters. (d) Embeddings taken from a jointly pretrained trunk, however, show structured yet overlapping clusters, suggesting positive representation transfer between molecules and materials has occurred**.



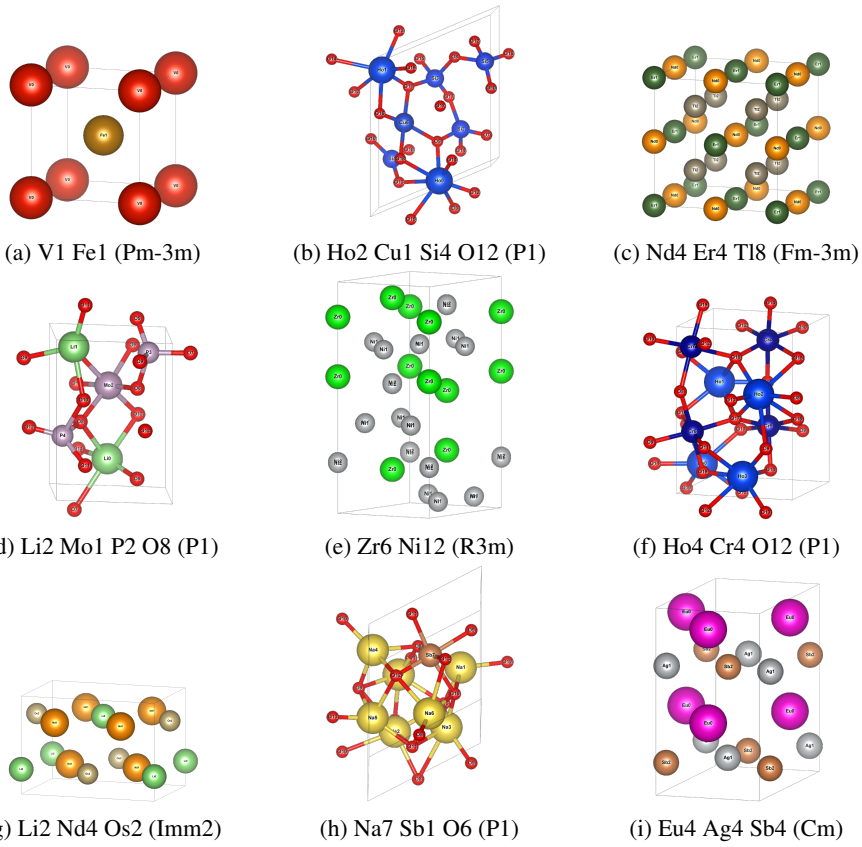
(a) Untrained model

(b) QM9-only pretrained model

(c) MP20-only pretrained model

(d) Jointly pretrained model

Figure 5: PCA plots of latent embeddings for carbon, nitrogen, oxygen, and fluorine atoms from ZATOM-1's Transformer trunk for 100 data points from the MP20 and QM9 datasets, as well as 100 molecules/materials each generated by ZATOM-1. Each point denotes an atom, with its shape representing its chemical system type, its size defined by whether it comes from real or generated data, and its color indicating its element type. **(a)** Embeddings taken from an untrained trunk are distributed in an overly clustered and chemically-segmented manner, with a quantitatively high crystal-molecule shift between atom embeddings of the same element and source type. **(b-c)** Embeddings taken from a trunk pretrained on either QM9 or MP20 form less well-structured molecule and material clusters, with a quantitatively higher intra-element spread and crystal-molecule shift. **(d)** A jointly pretrained model, however, produces trunk embeddings that achieve the best quantitative balance across all clustering and transfer learning metrics. Moreover, in such a model, principal component 1 largely relates to whether a system is a crystal (within range -2 – 4) or a molecule, while principal component 2, generally speaking, relates to an atom's type. **Joint pretraining induces a latent space that shows distinct clusters for different atom types, with oxygen (and, to some extent, fluorine) atoms having similar representations in both molecules and crystals. Notably, the overlap in oxygen atom representations suggests that ZATOM-1's latent space captures shared chemical properties across non-periodic and periodic systems, enabling effective knowledge transfer during joint pretraining.**



(a) V1 Fe1 (Pm-3m) (b) Ho2 Cu1 Si4 O12 (P1) (c) Nd4 Er4 Tl8 (Fm-3m)

(d) Li2 Mo1 P2 O8 (P1) (e) Zr6 Ni12 (R3m) (f) Ho4 Cr4 O12 (P1)

(g) Li2 Nd4 Os2 (Imm2) (h) Na7 Sb1 O6 (P1) (i) Eu4 Ag4 Sb4 (Cm)

Figure 6: Generated crystals from ZATOM-1 trained jointly on MP20 materials and QM9 molecules.

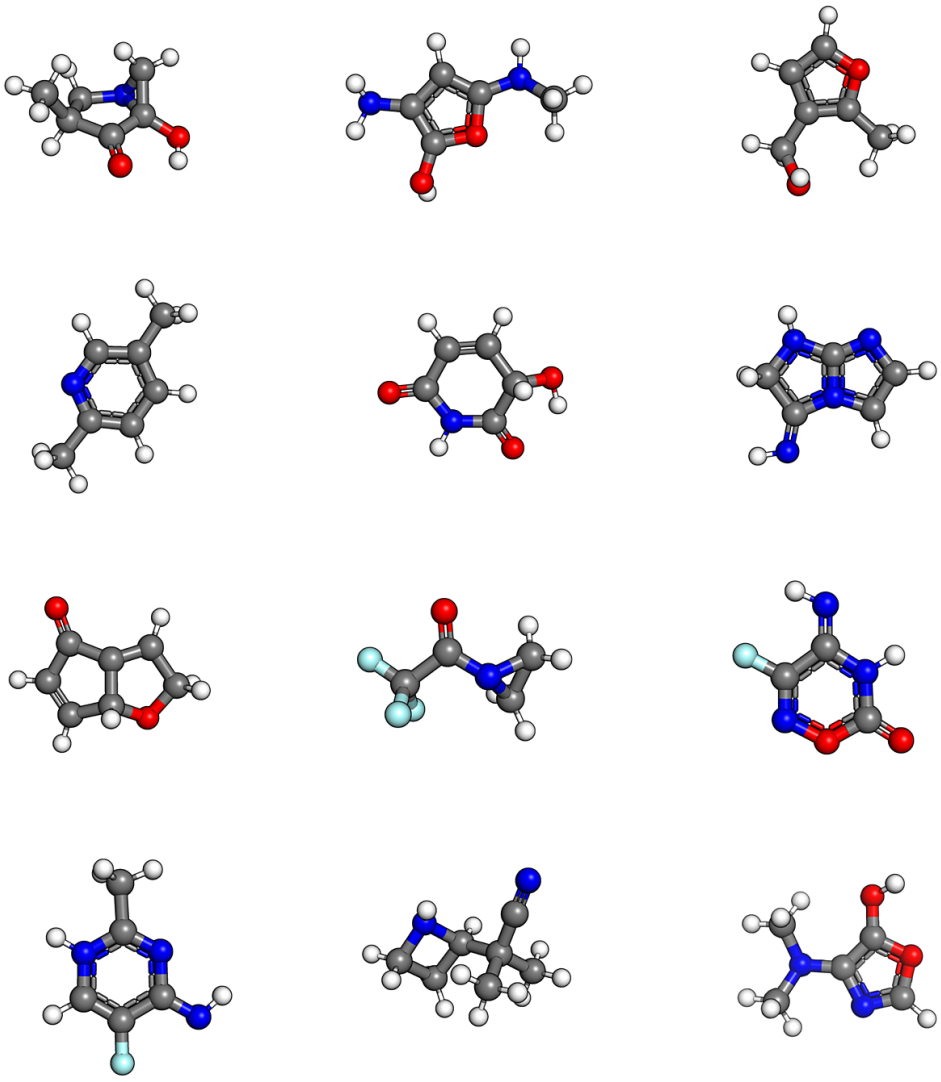
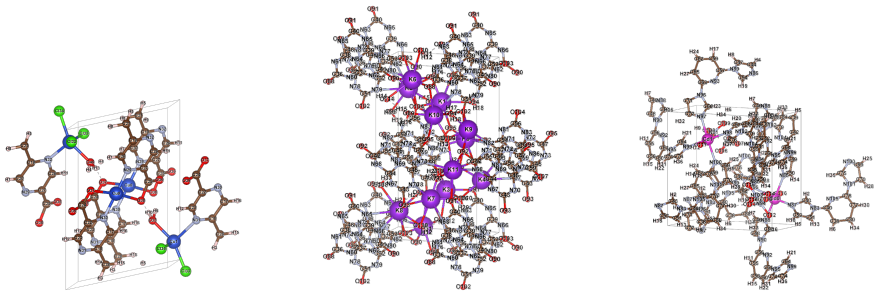
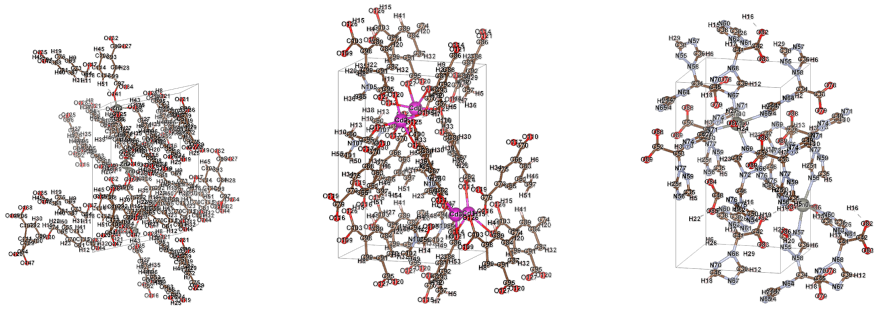


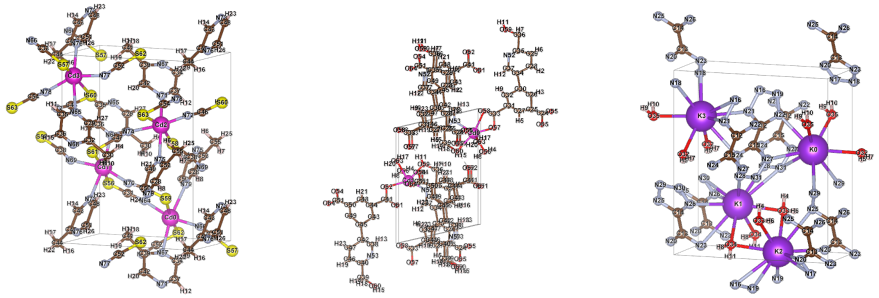
Figure 7: Generated molecules from ZATOM-1 trained jointly on MP20 materials and QM9 molecules.



(a) Cu3 H14 C12 N4 Cl2 O6 (P1) (b) K12 H24 C24 N28 O24 (P1) (c) Cd2 H36 C44 N24 O12 (P1)



(d) Zn4 H48 C64 O32 (P1) (e) Cd4 H52 C48 N4 O20 (P1) (f) Zn2 H28 C24 N24 O12 (P1)

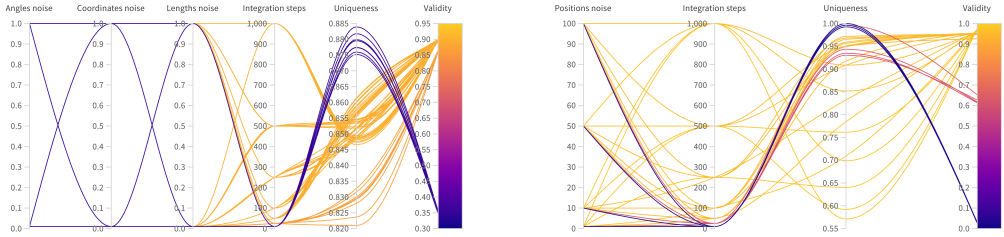


(g) Cd4 H24 C28 S8 N16 (P1) (h) Cd2 H22 C28 N2 O12 (P1) (i) K4 H8 C4 N16 O4 (P1)

Figure 8: Generated metal-organic frameworks from ZATOM-1 trained on QMOF.

D Additional Model Details

Hyperparameters. Table 5 reports the hyperparameters used to pretrain different versions of ZATOM-1 on MP20 and QM9, and Table 6 shows which hyperparameters were used to pretrain ZATOM-1 on QMOF or OMol25 as well as on GEOM-Drugs jointly with other datasets. Likewise, Table 7 displays the hyperparameters used to finetune ZATOM-1 (80M) for the downstream tasks explored in this work. For generative inference, Figure 9 shows the results of a hyperparameter sweep, where we find that using a limited number of integration steps and adding (varying levels of) white noise to each continuous modality during sampling generally improves sample validity and uniqueness. Notably, we find that for larger chemical systems, such as the molecules present in GEOM-Drugs, a smaller noise scale for atom positions (e.g., 0.01) is beneficial. Additionally, in all generative contexts, we unexpectedly find that classifier-free guidance worsens the quality of generated samples considerably (e.g., degrading 90% materials validity rates to 40%), motivating us to disable such inference guidance by default. We believe this may be addressed with alternative class label conditioning methods, which we defer to future work.



(a) Crystals – MP20 (b) Molecules – QM9

Figure 9: **Tuning inference hyperparameters for best performance.** Best generative modeling results for crystals and molecules, balancing sample validity and uniqueness, are generally achieved with a limited number of integration steps (e.g., $T = 100$), a high white noise scale γ_g (e.g., 50) for continuous, non-periodic atom positions, and a low noise scale (e.g., 0.01) for continuous, periodic modalities such as fractional coordinates.

Compute resources. All experiments used an internal SLURM cluster containing GPU nodes with 60 GBs of CPU memory and 80 GBs of (NVIDIA A100) GPU memory associated with each of the four GPUs attached to each node. In total, all experiments required ∼20,000 GPU hours for training and inference and 1 TB of local storage for datasets, model checkpoints, and experiment artifacts, with ∼3,000 GPU hours spent on preliminary experiments that did not make it into this manuscript.

Limitations. We discuss our method's lack of (e.g., 100M) dataset size scaling results and preliminary precision for energy and force modeling in Section 4 and Appendix F.4, respectively. Meaningful additional experiments in these two areas require a significantly larger compute budget than is available for this work. We also note that our method's molecular and material property prediction results show room for improvement compared to single-task learning baselines, which could be addressed with either dedicated single-task learning experiments for each molecular/material property (n.b., out of scope for this methodologically-focused work, due to compute constraints) or with more expressive variations of flow matching-based pretraining (n.b., which we aim to explore in future work). Importantly, our current computational experiments in Section 4 and Appendix F already verify the effectiveness and computational efficiency of our proposed method in a fair (i.e., data, task, and compute-controlled) setting, with future wet-lab experiments in preparation.

Broader Impacts This paper presents work intended to advance the field of Scientific Machine Learning. There are many potential societal consequences of our work, including the acceleration of scientific discovery in the molecular and materials sciences. We acknowledge the risk that, in the hands of "bad actors", extensions of such technologies may be used with harmful ends in mind. However, with strong research community guidelines in place, such as those of `https://responsiblebiodesign.ai/`, it is our hope that efforts in developing scientific foundation models for the chemical sciences will disproportionately influence the positive societal outcomes of such research, such as improved medicines and energy technologies, as opposed to possible negative consequences, such as the development of new bioweapons.

Licensing. ZATOM-1 is available under a permissive BSD license. The license of each publicly available dataset that ZATOM-1 uses for pretraining or finetuning is listed below.

1. QM9, found at `https://pytorch-geometric.readthedocs.io/en/stable/generated/torch_geometric.datasets.QM9.html`, has a Creative Commons Attribution 4.0 (CC-BY-4.0) license.
2. MP20, found at `https://huggingface.co/datasets/chaitjo/MP20_ADiT`, has an MIT license.
3. GEOM-Drugs, found at `https://bits.csb.pitt.edu/files/geom_raw/`, has a Creative Commons Attribution 4.0 (CC-BY-4.0) license.
4. QMOF, found at `https://huggingface.co/datasets/chaitjo/QMOF150_ADiT`, has an MIT license.
5. Matbench, found at `https://huggingface.co/datasets/Ty-Perez/matbench_properties`, has an MIT license.
6. OMol25, found at `https://huggingface.co/facebook/OMol25`, has a Creative Commons Attribution 4.0 (CC-BY-4.0) license.
7. MPtrj, found at `https://huggingface.co/datasets/nimashoghi/mptrj`, has an MIT license.

Table 5: Hyperparameters across different model versions when jointly pretraining on QM9 & MP20.

Hyperparam.	**ZATOM-1-WD (80M)**	**ZATOM-1 (80M)**	**ZATOM-1-L (160M)**	**ZATOM-1-XL (300M)**
# Params.	77,350,264	77,350,264	162,615,544	293,887,096
Hidden size	512	512	768	1024
# Transformer blocks	16	16	16	16
# Attn. heads	8	8	8	8
Train t distribution	Beta(1.8,1)	Beta(1.8,1)	Beta(1.8,1)	Beta(1.8,1)
$\lambda_{\text{discrete}}$	0.1	0.1	0.1	0.1
Learning rate	0.001	0.001	0.0005	0.00025
Learning rate schedule	N/A	N/A	N/A	N/A
Optimizer	AdamW	AdamW	AdamW	AdamW
Weight decay	0.01	0.0	0.0	0.0
EMA weight	0.999	0.999	0.999	0.999
Batch size	64	256	128	64
# Grad. accum. steps	1	1	2	4
# Rotation augs.	8	8	8	8
Effective batch size	2,048	32,768	32,768	32,768
# GPUs	4 (80GB)	16 (80GB)	16 (80GB)	16 (80GB)
Training duration	10h	24h	48h	96h
# Sampling steps	100	100	100	100
$g(t)$	$\frac{1}{t+0.01}$	$\frac{1}{t+0.01}$	$\frac{1}{t+0.01}$	$\frac{1}{t+0.01}$
γ_g ($\rightarrow$ for small molecule coordinates)	0.01 ($\rightarrow$ 50.0)	0.01 ($\rightarrow$ 50.0)	0.01 ($\rightarrow$ 50.0)	0.01 ($\rightarrow$ 50.0)

Table 6: Hyperparameters when jointly pretraining ZATOM-1 on GEOM-Drugs, QMOF, or OMol25.

Hyperparam.	**GEOM-Drugs & MP20**	**QMOF**	**OMol25**
# Params.	77,350,264	77,350,264	77,350,264
Hidden size	512	512	512
# Transformer blocks	16	16	16
# Attn. heads	8	8	8
Train t distribution	Beta(1.8,1)	Beta(1.8,1)	Beta(1.8,1)
$\lambda_{\text{discrete}}$	0.1	0.1	0.1
Learning rate	0.001	0.001	0.001
Learning rate schedule	N/A	N/A	N/A
Optimizer	AdamW	AdamW	AdamW
Weight decay	0.0	0.0	0.0
EMA weight	0.999	0.999	0.999
Batch size	32	32	16
# Grad. accum. steps	1	1	1
# Rotation augs.	8	8	8
Effective batch size	4,096	4,096	2,048
# GPUs	16 (80GB)	16 (80GB)	16 (80GB)
Training duration	168h	672h	168h
# Sampling steps	100	100	100
$g(t)$	$\frac{1}{t+0.01}$	$\frac{1}{t+0.01}$	$\frac{1}{t+0.01}$
γ_g	0.01	0.01	0.01

Table 7: Finetuning hyperparameters across different downstream prediction tasks for ZATOM-1 (80M), optionally trained in sequential order.

Hyperparam.	**Mol. Properties (80M ❄ + 20M 🔥) →**	**Mol. / Mat. Properties (80M ❄ + 20M 🔥) →**	**Mol. / Mat. Energy & Forces (80M ❄ + 220M 🔥)**
# Dedicated params.	18,104,595	18,104,595	217,255,140
Aux. hidden size	512	512	1024
# Aux. transformer blocks	4	4	8
# Attn. heads	8	8	8
Train t distribution	max(Beta(1.8,1), 0.98)	max(Beta(1.8,1), 0.98)	max(Beta(1.8,1), 1.0)
λ_{forces}	N/A	N/A	5.0
Learning rate	0.001	0.001	0.0003
Learning rate schedule	N/A	N/A	N/A
Optimizer	AdamW	AdamW	AdamW
Weight decay	0.0	0.0	0.0
EMA weight	N/A	N/A	N/A
Batch size	128	32	72
# Grad. accum. steps	2	4	1
# Rotation augs.	8	8	1
Effective batch size	4,096	2,048	1,152
# GPUs	2 (40GB)	2 (80GB)	16 (80GB)
Training duration	48h	192h	120h

E PLATOM-1 – Explicitly Equivariant Transformer Architecture

A long-standing discussion in the molecular modeling community revolves around whether symmetries in the data should be learned or hard-coded into the model architecture. Given the considerable success of entirely non-equivariant models like AlphaFold 3 [1], the community's consensus has shifted toward the conclusion that equivariant models would be slower, harder to optimize, and generally inferior compared to architectures without such inductive biases. This appendix provides empirical evidence challenging this common belief, showing that – *when done right* – equivariance can unlock significant performance gains.

Section E.1 introduces the Trunk-based Flow Platoformer (TFP), a rotation equivariant counterpart of the Trunk-based Flow Transformer (TFT) architecture underlying ZATOM-1. Empirical results, showing faster convergence and higher validation scores of TFP, are presented in Section E.2 and Figure 10. For now, we focus exclusively on molecular tasks, since the current fractional coordinate-based invariant embedding of materials used in ZATOM-1 is at odds with the equivariant inputs expected by TFP.

E.1 Trunk-based Flow Platoformer architecture (TFP)

The Trunk-based Flow Platoformer (TFP) is a rotation equivariant counterpart of our TFT architecture from Section 2.2, replacing its standard transformer layers with *Platonic Transformer* layers introduced by Islam et al. [44]. These layers are equivariant to discrete subgroups of roto-reflections $G < \mathrm{O}(3)$ which are symmetries of Platonic solids like the tetrahedron ($|G| = 12$) or cube ($|G| = 24$) [15].

Rather than relying on $\mathrm{O}(3)$-irrep features (i.e., $\mathrm{O}(3)$-irreducible representations), as most equivariant point cloud message passing neural networks do, the features of TFP are associated with group elements, essentially encoding molecular geometry as viewed from $|G|$ different frames of reference. Mathematically, this is known as a *regular G-representation* [108]. In code, a point cloud of N features with C channels is represented by a tensor of shape $(N, |G|, C)$ rather than the usual shape (N, C). Such regular representation-valued features are more commonly used in vision models, most notably group convolutional networks [20, 107, 5].

TFP architecture. Before applying Platonic layers, inputs need to be lifted to regular G-representations. Scalars (e.g., atom types) are lifted to features that are invariant over the group axis. Vector-valued inputs, like Euclidean coordinates, lift to non-trivial features encoding directional information. Platonic linear layers satisfy a symmetry constraint which reduces their parameter count via $|G|$-fold weight sharing. The original linear embedding layers of TFT are replaced by such Platonic linear layers, applied after G-lifts.

Platonic attention layers utilize common scaled dot-product attention [98], and are thus compatible with IO-efficient Flash Attention kernels [22]. The group axis of regular G-features is thereby reshaped into the attention heads axis rather than channels, i.e., $(H \cdot |G|,\ N,\ C)$ instead of $(H,\ N,\ |G| \cdot C)$ – this has the advantage that attention scores remain regular representation-valued rather than becoming G-invariant due to contracting over G [44]. We use full softmax attention instead of the linear attention variants explored in the original Platonic Transformer paper, as these were required mainly to address training instabilities when encoding rotary positions in molecular modeling tasks.

A core component of the Platonic Transformer is a G-equivariant generalization of Rotary Position Embeddings (RoPE) [93], which guarantees both translational and G-equivariance, i.e., approximate $\mathrm{E}(3)$-equivariance. As we aim to stick as close as possible to the original TFT design without *relative* RoPE attention, we omit this feature, relying merely on the *absolute* embedding of Euclidean coordinates via lifting and Platonic linear layers discussed above. Note that this design choice breaks translational symmetries but preserves G-equivariance. Since we zero-center non-periodic 3D atom coordinates, this is not an issue in practice.

We reimplement and extend many other operations from the original Platonic Transformer code, including cross-attention layers, new normalization modules, embedding layers, and optimized Platonic linear layers. Given these modules, we implement TFP in analogy to the TFT design

	Linear	G-linear	Attention	G-attention
Compute FLOPS	$\mathcal{O}(ND^2)$	$\mathcal{O}(N\|G\|^2C^2)$	$\mathcal{O}(N^2D)$	$\mathcal{O}(N^2\|G\|C)$
Memory	$\mathcal{O}(ND)$	$\mathcal{O}(N\|G\|C)$	$\mathcal{O}(ND)$	$\mathcal{O}(N\|G\|C)$
Parameters	D^2	$\|G\|\cdot C^2$	—	—

Table 8: **Compute, memory, and parameter complexity of TFT and TFP.** TFT and TFP operate on tensors of shapes $(N, |G|, C)$ and (N, D), respectively. Memory costs follow immediately from these shapes. Linear layers in TFT come with the usual quadratic number of FLOPS and parameters in their D channels. TFP is internally leveraging `nn.Linear` layers as well, however, after flattening features to shape $(N, |G|\cdot C)$, implying a computational cost quadratic in $|G|$ as well. Equivariance imposes a symmetry constraint on the matrices (weight sharing), resulting in a $|G|$ times-enhanced parameter efficiency. TFT's attention layers have the usual computational complexity linear in channels D, while those of TFP are linear in $|G|C$. Neither introduces additional parameters.

summarized in Algorithm 1. Final predictions are projected back from G-valued features to scalar and vector outputs. Our implementation is available as part of the ZATOM-1 code repository.

Compute and parameter cost. As TFP operates on tensors of shape $(N, |G|, C)$ with an additional group axis, one may wonder how its computational, memory, and parameter complexities compare to those of the TFT baseline. Given that regular features have $|G|C$ effective dimensions, it is common practice to reduce their channel count C relative to the D channels of TFT's shape (N, D) features. The implied complexities for TFT's and TFP's linear and attention layers are summarized in Table 8. From these complexities, the following two heuristics to choose C to build "matching" architectures arise [106]:

Compute-matching: To match the compute and memory cost of TFP and TFT, one sets $C = D/|G|$, essentially sticking with the same effective number of channels but splitting them into group and channel axes. However, this implies G times fewer trainable parameters in TFP.

Parameter-matching: Alternatively, one may match the number of parameters of TFP and TFT by setting $C = D/\sqrt{|G|}$. This leads to a $|G|$ fold increase in memory and Floating-point Operations Per Second (FLOPS).

As a middle ground between these two extremes, we choose $C = D/|G|^{2/3}$, resulting in $\sqrt{|G|}$ fewer parameters and $\sqrt{|G|}$ increased computational cost. For the tetrahedral group, $|G| = 12$ and $\sqrt{|G|} \approx 3.5$, such that a TFP analog to ZATOM-1 with 80M parameters has 23M parameters. Note that this model happens to be approximately compute-matched with ZATOM-1-XL, which has 300M parameters, i.e., 3.75 times the number of ZATOM-1's parameters.

E.2 Empirical results

Based on TFP, we implement PLATOM-1, the Platonic group equivariant counterpart of ZATOM-1. Due to limited computational resources, we specifically evaluate the (orientation-preserving) tetrahedral group with $|G| = 12$ rotations (no reflections) and 23M parameters as discussed above. Since PLATOM-1 is only equivariant w.r.t. discrete rotations, we use the same continuous $\mathrm{SO}(3)$ rotational data augmentation as used for ZATOM-1.

QM9 molecule generation. We first compare to the QM9-only pretrained ZATOM-1 baseline flow model in Figure 10. It is evident from these plots that PLATOM-1 converges much faster than ZATOM-1 in terms of training and validation losses, as well as validation metrics. Such improved convergence is common for equivariant models, as they do not need to implicitly learn symmetries from data [108, 123]. The best TFP checkpoint on the validation set is reached after 249 epochs, instead of 399 epochs for TFT, partially offsetting TFP's increased computational cost.

Table 9 confirms that these findings translate to all test metrics calculated for these checkpoints. In particular, PLATOM-1 achieves better results than ZATOM-1, prior equivariant QM9-only pretrained

models Equivariant Diffusion [41] and Symphony [21], and the much larger and slower latent diffusion model ADiT [49]. It even outperforms our 300M parameter model ZATOM-1-XL, despite the latter being jointly trained on molecules and materials.

Joint molecule and material pretraining. One of the main hypotheses of ZATOM-1 is the benefit of positive transfer learning when pretraining jointly on molecular and material modalities. A similar mutual benefit should, in principle, be observable for PLATOM-1. However, our current encoding of materials is inherently incompatible with Platom-1's equivariant design, leading to training instabilities. Specifically, materials are not represented in terms of their (vector-valued) Euclidean coordinates, but rather in terms of fractional coordinates and lattice basis lengths and angles. All these quantities are scalars, and therefore lift to features that are invariant over the G-axis. Being blind to the original Euclidean geometry expected by PLATOM-1, the model fails to converge.

While ZATOM-1 can learn to leverage fractional coordinates inputs, we believe that this is not the ideal (equivariant) data representation for PLATOM-1, since it obfuscates the Euclidean geometry of the materials. As part of future work, we plan to experiment with TFP using a unified Euclidean coordinate representation of molecules and materials, for which we expect to see positive transfer learning results in a PLATOM-1-style model.

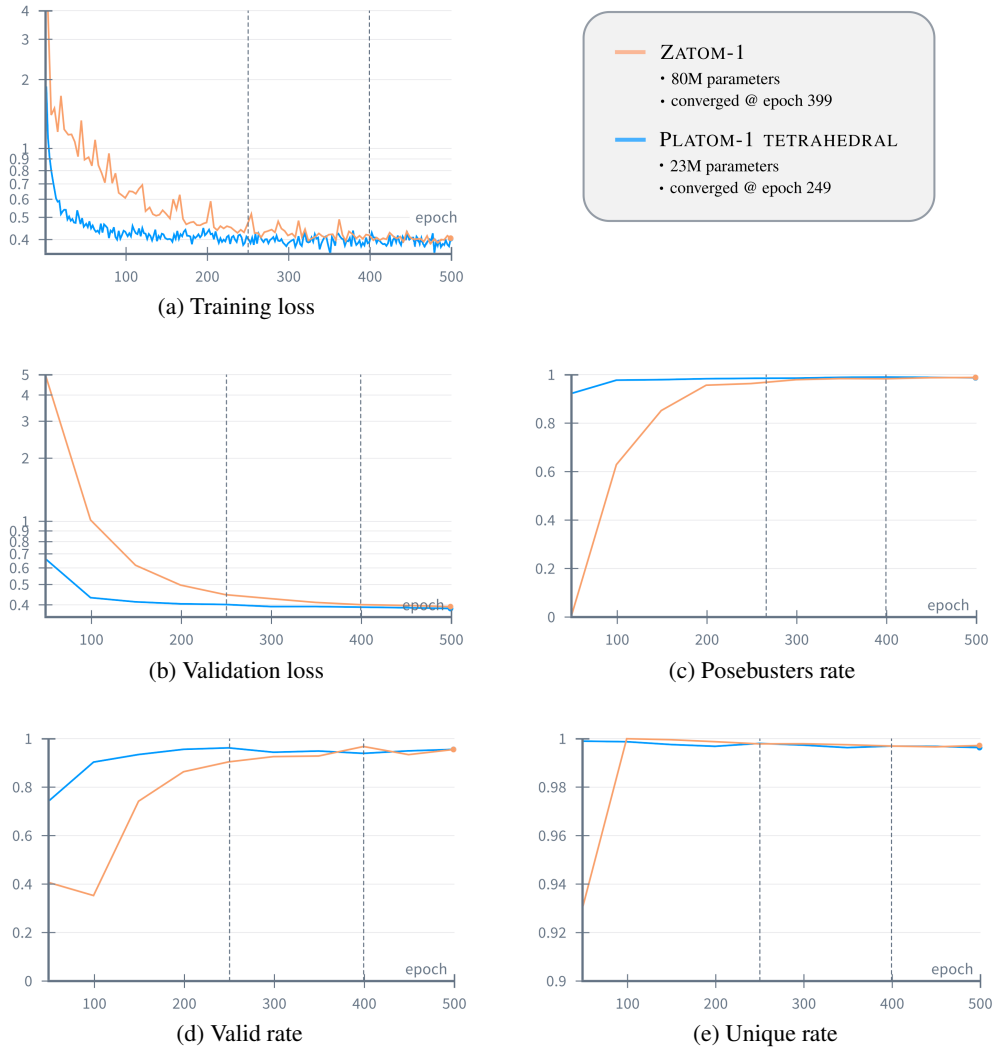
Table 9: **Molecule generation results on QM9 with explicit equivariance.** We report (a) validity and uniqueness rates as well as (b) % pass rates on 7 sanity checks from PoseBusters for 10,000 sampled molecules. All models explicitly generate hydrogen atoms. Notably, QM9-only PLATOM-1 achieves its results using 249 training epochs compared to QM9-only ZATOM-1 (80M), which requires 399 training epochs to converge. Despite its smaller size and not being jointly pretrained on molecules and materials, PLATOM-1 is competitive with ADiT and ZATOM-1-XL.

(a) **Validity results**

Model	# Params	Validity (%) ↑	Unique (%) ↑
One-stage training			
Equivariant Diffusion	20M	91.90	98.69
Symphony	2M	83.50	97.98
Two-stage training			
GeoLDM	20M	93.80	**98.82**
QM9-only ADiT	180M	92.19	97.90
Jointly trained ADiT	180M	94.45	97.82
Ours: One-stage training			
QM9-only ZATOM-1	80M	92.88	97.71
QM9-only PLATOM-1	23M	95.20	98.13
Jointly trained ZATOM-1	80M	94.94	97.16
Jointly trained ZATOM-1-L	160M	**95.26**	96.84
Jointly trained ZATOM-1-XL	300M	95.19	97.10

(b) **PoseBusters results**

Test (% pass) ↑	Symphony (QM9-only)	Eq. Diff. (QM9-only)	ADiT (jointly trained)	PLATOM-1 (QM9-only)	ZATOM-1 (QM9-only)	ZATOM-1 (jointly trained)
Atoms connected	99.92	99.88	99.70	99.97	99.95	**99.98**
Bond angles	99.56	**99.98**	99.85	99.94	99.90	99.95
Bond lengths	98.72	**100.00**	99.41	99.96	99.89	99.97
Aromatic ring flat	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
Double bond flat	99.07	98.58	99.98	**100.00**	**100.00**	99.99
Internal energy	95.65	94.88	95.86	99.73	99.63	**99.78**
No steric clash	98.16	99.79	99.79	**99.84**	99.81	99.81



(a) Training loss

(b) Validation loss

(c) Posebusters rate

(d) Valid rate

(e) Unique rate

Figure 10: **QM9-only pretraining of ZATOM-1 vs. PLATOM-1.** These plots show the training dynamics of ZATOM-1 compared to the analogous tetrahedral group equivariant PLATOM-1 model described in Section E.1. Due to its equivariance, the Platonic model does not explicitly need to learn symmetries, resulting in significantly faster convergence while achieving improved evaluation metrics. These findings translate to the test results in Table 9.

Table 10: **Metal-organic framework generation results on QMOF.** We report sanity checks from MOFChecker [48] for 1,000 sampled MOFs, comparing methods trained only on QMOF for at least 10,000 epochs. (↑/↓ indicate higher/lower is better, respectively, and * denotes a model that has not converged after 50,000 training epochs)

Test%	QMOF ADiT	QMOF ZATOM-1*
Has carbon ↑	100.0	100.0
Has hydrogen ↑	99.6	100.0
Has atomic overlap ↓	8.3	8.8
Has overcoord. C ↓	23.6	1.1
Has overcoord. N ↓	1.5	0.0
Has overcoord. H ↓	1.0	2.9
Has undercoord. C ↓	60.0	65.4
Has undercoord. N ↓	39.1	22.8
Has undercoord. rare earth ↓	0.4	0.0
Has metal ↑	100.0	100.0
Has lone molecule ↓	72.9	80.5
Has high charge ↓	0.9	0.5
Has suspicious terminal oxo ↓	2.6	0.3
Has undercoord. alkali ↓	1.0	0.3
Has geom. exposed metal ↓	7.0	1.8
Validity rate (all passed) ↑	15.7	15.1

Table 11: **Molecule generation results on OMol25.** Validity, uniqueness, and % pass rates on PoseBusters for 16,000 sampled molecules. ZATOM-1, pretrained on the (non-equilibrium) molecules of OMol25, takes a first step toward non-equilibrium 3D molecule generation.

Metric (% pass) ↑	OMol25 ZATOM-1
Validity	30.4
Uniqueness	93.5
Atoms connected	17.0
Bond angles	78.9
Bond lengths	54.6
Aromatic ring flat	100.0
Double bond flat	100.0
Internal energy	73.5
No steric clash	60.2
PoseBusters valid	15.1

F Additional Results

F.1 Exploring metal-organic framework generation

Table 10 investigates ZATOM-1's ability to generate metal-organic frameworks (MOFs), a novel class of materials with several promising applications [53]. With minimal hyperparameter tuning, we find that ZATOM-1 can match the MOF generation validity rate of ADiT despite the numerous chemical and geometric constraints of MOFs, such as requiring a balance of metal, carbon, and hydrogen atoms without steric clashes. Nonetheless, the model (as currently evaluated) has not converged, indicating that its proportion of successfully generated MOFs could likely be improved with further training.

F.2 Stepping towards non-equilibrium 3D molecule generation

Table 11 studies ZATOM-1's ability to generate non-equilibrium molecules, like those contained in the OMol25 dataset. With minimal hyperparameter tuning and only 80 epochs of OMol25-based pretraining (due to compute resource constraints), we find that ZATOM-1 establishes a strong first step towards non-equilibrium 3D molecule generation. However, we note that it is currently unclear whether using the default settings for RDKit and the PoseBusters software suite is an appropriate way of assessing non-equilibrium molecule validity and realism, e.g., as PoseBusters' unit tests have likely been tuned using a different distribution of 3D molecules. Nonetheless, with additional training and hyperparameter/metric tuning, these results could likely be improved, which we leave for future work.

Table 12: **Property prediction for 3D molecules (QM9) / materials (Matbench) with joint molecule and materials finetuning.** Results are reported as *test* mean absolute errors (for ZATOM-1, with standard deviations over three runs with different random seeds). † denotes using different data partitions. / represents a task for which both (QM9) molecule and (Matbench) material property labels are available. NN denotes pretraining using a Noisy Nodes self-supervised objective [118]. The best (or second-best) results for each optimized (i.e., hyperparameter-tuned) or unoptimized (i.e., non-hyperparameter-tuned) category are in **bold** (or underlined). Notably, finetuning jointly for multi-task molecule and ($\Delta\varepsilon$) materials property prediction improves or marginally trades off ZATOM-1's molecule property prediction performance for expanded applicability of its property predictions to both molecules and materials simultaneously.

Model / Metrics ↓	# Params	α (a_0^3)	$\Delta\varepsilon$ (meV)	$\varepsilon_{\text{HOMO}}$ (meV)	$\varepsilon_{\text{LUMO}}$ (meV)	μ (D)	C_ν (cal/mol K)	G^{ATOM} (meV)	H^{ATOM} (meV)	R^2 (a_0^2)	U^{ATOM} (meV)	U_0^{ATOM} (meV)	$ZPVE$ (meV)
Optimized single-task learning													
DimeNet++†	5M	.044	33.0 / **199**	25	20	.030	.023	8.00	7.00	.331	6.00	6.00	1.21
EGNN†	1M	.071	48.0 / ----	29	25	.029	.031	12.0	12.0	.106	12.0	11.0	1.55
PaiNN	1M	.045	46.0 / ----	28	20	.012	.024	**7.35**	**5.98**	.066	**5.83**	**5.85**	1.28
TorchMD-NET	7M	.059	36.0 / ----	20 (16 w/ NN)	18 (13 (w/ NN)	.011	.026	7.62	6.16	**.033**	6.38	6.15	1.84
SphereNet	2M	.046	32.0 / ----	23	18	.026	**.021**	8.00	6.00	.292	7.00	6.00	**1.12**
SEGNN†	1M	.060	42.0 / ----	24	21	.023	.031	15.0	16.0	.660	13.0	15.0	1.62
EQGAT	-	.053	32.0 / ----	20	16	.011	.024	23.0	24.0	.382	25.0	25.0	2.00
Equiformer	4M	.046	30.0 / ----	15	14	.011	.023	7.63	6.63	.251	6.74	6.59	1.26
EquiformerV2	11M	.050	**29.0** / ----	**14**	**13**	.010	.023	7.57	6.22	.186	6.49	6.17	1.47
PÖNITA	-	**.038**	30.4 / ----	16	15	.012	.024	8.63	8.04	.235	8.67	8.31	1.29
Platonic Transformer	-	.049	37.4 / ----	22	17	**.010**	.024	12.0	12.0	.222	11.9	13.0	1.30
Unoptimized single-task learning													
AIM-STL†	-	.181	---- / ----	61	54	**.067**	.072	66.2	63.9	**.503**	64.2	58.8	**4.54**
Unoptimized multi-task learning													
AIM-MTL-Scalar†	-	.268	---- / ----	59	71	.089	.103	113	112	4.18	112	112	9.82
AIM-MTL-Matrix†	-	.251	---- / ----	61	72	.088	.103	112	118	4.07	116	116	12.2
QM9-pretrained ZATOM-1 (OURS)	80M ❄ + 20M 🔥	.105±.000	52.0±0.1 / 362±1	37±0.1	36±0.1	.102±0.001	.054±0.000	47.2±0.2	48.5±0.2	3.93±0.00	48.6±0.2	48.7±0.2	5.03±0.01
Jointly pretrained ZATOM-1 (OURS)	80M ❄ + 20M 🔥	**.095±.000**	**46.2±0.1 / 354±1**	**34±0.1**	**32±0.1**	.087±.000	**.047±.000**	**39.2±0.2**	**42.1±0.2**	3.33±0.01	**41.6±0.2**	**41.6±0.2**	4.62±0.02

Table 13: **Energy and force prediction for 3D materials (MPtrj) / molecules (OMol25-4M).** The results are reported as *validation* Energy (in meV or meV/Atom) and Force (in meV/Å) mean absolute error (MAE). All models directly predict forces, rather than using the gradient of their predicted energies with respect to input 3D coordinates for conservative force prediction, to considerably increase their computational efficiency. * denotes a model that was pretrained on OMol25 and which has not converged after 80 training epochs.

Model	# Params	Energy MAE/Atom (meV) ↓	Energy MAE (meV) ↓	Force MAE (meV/Å) ↓
Material Interatomic Potential Prediction on MPtrj				
Non-pretrained Orb-v1	∼30M	-	350.00	122.00
Denoising-pretrained Orb-v1	∼30M	-	210.00	32.00
Jointly pretrained ZATOM-1	80M ❄ + 220M 🔥	112.81	2631.22	30.92
Molecule Interatomic Potential Prediction on OMol25-4M				
eSEN-md-d.	∼51M	1.32	77.13	6.78
Molecule-only MLIP-pretrained ZATOM-1*	∼93M	7.13	514.00	86.99
Molecule-only (generatively-)pretrained ZATOM-1*	80M ❄ + 220M	19.46	1307.23	72.40
Jointly pretrained ZATOM-1	80M ❄ + 220M 🔥	34.45	2512.78	62.99

F.3 Balancing multi-task molecule and material property prediction

Table 12 presents the results of performing a second stage of finetuning for multi-task molecule and ($\Delta\varepsilon$) materials property prediction using QM9 and Matbench, starting from ZATOM-1's first stage of QM9 finetuning for multi-task molecule property prediction. While improving or marginally exchanging ZATOM-1's molecule property prediction performance, these results demonstrate that (generative) jointly pretrained ZATOM-1's *multi-task* property predictions can effectively be applied to both 3D molecules and materials, to our knowledge a *first-of-its-kind* result. In contrast, for materials property prediction in Table 4 of the main text, we report zero-shot $\Delta\varepsilon$ results for the model weights performing best specifically for multi-task molecule property prediction, to illustrate the broad applicability of ZATOM-1's pretrained molecule and material embeddings.

F.4 Extension to energies and forces

Table 13 explores ZATOM-1's ability to predict energies and forces to serve as an MLIP. Through this preliminary set of experiments, we find that the performance of jointly pretrained ZATOM-1's material force predictions for MPtrj slightly surpasses that of Orb-v1, a state-of-the-art, pretrained model for this dataset. Results for OMol25 demonstrate that jointly pretrained ZATOM-1 can balance the performance of its force predictions for molecules and materials jointly, highlighting its potentially broad applicability in molecular and materials sciences. Despite these promising first steps, ZATOM-1, when pretrained solely as an MLIP using OMol25, achieves considerably better energy predictions for OMol25 than jointly pretrained ZATOM-1, suggesting the presence of negative (energy-specific)

Table 14: QM9 molecule generation results after 20 epochs of QM9 multi-task property prediction finetuning.

Fine-Tuning Strategy	Validity ↑	Uniqueness ↑	Novelty ↑	PoseBusters Valid ↑
Trunk Unfreezing	0.17	0.33	0.33	0.00
LoRA	0.48	1.00	1.00	0.61
Trunk Freezing (Baseline)	0.95	0.97	0.96	0.99

Table 15: Comprehensive GEOM-Drugs molecule generation results. – denotes unavailable results, and * indicates results were estimated from 1,000 samples per Vonessen et al. [103].

Metric	ADiT	TABASCO	ZATOM-1 (Ours)	GEOM-Drugs
Validity ↑	0.95	0.98	0.94	–
Uniqueness ↑	1.00	0.99	1.00	–
Diversity ↑	0.91*	0.89	0.89	0.90
Novelty ↑	0.97*	0.99	0.98	0.00
QED ↑	–	0.64	0.64	0.67
Lipinski ↑	–	4.90	4.80	5.00
LogP ↑	–	2.83	2.77	2.92
Strain Energy ↓	46.36*	17.50	22.58	–
PoseBusters Valid ↑	0.85	0.92	0.94	0.94

transfer learning between jointly pretrained ZATOM-1's small-sized QM9 and MP20 pretraining examples and OMol25's much larger pretraining molecules (n.b., as the energy of an atomic system is a function of the system's size). To characterize this observation further, using the generative modeling weights studied in Table 11, we generatively pretrained ZATOM-1 and finetuned MLIP task-specific weights using OMol25 end-to-end, finding that *generative* pretraining with OMol25 leads to better force predictions than *MLIP* pretraining/finetuning with OMol25 (though energy predictions are improved with the latter approach). As a potential future direction, *joint* molecule and material generative pretraining (beyond 80 epochs [59], up to convergence) using more structurally diverse datasets, such as LeMat-Bulk and OMol25, could improve ZATOM-1's ability to balance energy and force predictions effectively; we defer this to future work due to compute resource constraints.

F.5 Finding an optimal method of finetuning

In our initial approach to finetuning, we tried unfreezing ZATOM-1's entire pretrained Transformer trunk for downstream tasks, which led to significant generative performance degradation. Subsequently, we tried addressing this issue by adopting Low-Rank Adaptation (LoRA) for Transformer trunk finetuning [42]. In this setting, we used rank $r = 8$, $\alpha = 16$, and a scaling factor of $\frac{\alpha}{r}$, targeting the Transformer trunk's query, key, value, and output projection layers (totaling 500k parameters). As shown in Table 14, whole-trunk unfreezing results in catastrophic failure regarding 3D molecule generation validity as verified by RDKit/PoseBusters. Although LoRA remains significantly more robust in this regard than trunk unfreezing, we find that both trunk unfreezing and LoRA show further generative performance decay over extended finetuning. These findings support our current hypothesis: maintaining generative integrity while performing representation learning requires finetuning downstream task-specific weights rather than modifying the pretrained Transformer trunk.

F.6 Comprehensively examining GEOM-Drugs molecule generation performance

Table 15 reports GEOM-Drugs molecule generation results for 10,000 samples across several additional metrics: Diversity and Novelty [103]; Quantitative Estimate of Drug-likeness (QED), Lipinski's Rule of 5, and Crippen's LogP [12]; and PoseCheck's Strain Energy [36]. As shown in Table 15, ZATOM-1 generates the most physically realistic molecules, matching the GEOM-Drugs reference dataset in PoseBusters validity (0.94). While TABASCO exhibits a slight advantage in chemical realism—specifically within Validity, Lipinski, and LogP scores—ADiT lags significantly in physical realism. Further, both ZATOM-1 and TABASCO achieve competitive Strain Energy bounds compared to ADiT. Overall, these results demonstrate that ZATOM-1 excels at producing physically realistic, drug-like molecules with chemical properties that closely mirror those of the GEOM-Drugs dataset.